



UNIVERSITY OF OSLO

Department of Informatics

**The Effect of User
Involvement on System
Acceptance:**

**The Development,
Maintenance and Drifting
of a Groupware System**

Master Thesis

(60 credits)

Joar Wisth Paulsen

02. May 2008

Abstract

Every large groupware project includes by necessity many different users. The top-down approach of the 80s and 90s is changing towards a more user centered approach. What implications does a focus on user involvement have on system acceptance of groupware systems? By looking at Jonathan Grudin's "Eight Challenges for Developers", this thesis will analyze a modern groupware system compared to his recommendations for more user involvement. Some social issues will be analyzed within social capital theory, and the thesis is written within the MOSCITO Project. My study shows positive results of user involvement, but also the importance of management involvement and a good structured process. In accordance with some of Grudin's views, some groupware traps are avoided by integration into the everyday system. A positive work community may also be a factor of failure and success on system acceptance.

Acknowledgements

The writing of this thesis has been a very interesting process. I have gained valuable experience in doing field studies and analyzing quantitative data, and also the task of putting together a large project. The work has been fun, stressing, rewarding and straining. It is with great joy and pride I deliver this paper, and though it is good to be done I think I will miss this period of my life.

I want to thank some people:

My teaching supervisor Jo Herstad for good advices and great interest for my subject.

Andreas for hooking me up with my case study.

All the people in the company where I conducted my research, who have been very helpful with setting up interviews and also seemed genuinely interested in my case.

The people in the MOSCITO project for good feedback and valuable experience.

Linda for coffee, motivation and corrections on my text.

Thanks!

Contents

Abstract.....	3
Acknowledgements	5
1 Introduction.....	11
1.1 Motivation.....	11
1.2 Research Question	11
1.3 Problem Area	12
1.4 Background	15
1.5 Synopsis.....	17
2 Background Theory	19
2.1 CSCW	19
2.1.1 History.....	19
2.1.2 Definition	21
2.1.3 Complexities	23
2.1.4 Different types of technology.....	24
2.1.5 Groupware.....	24
2.2 Social Capital.....	25
2.2.1 Definition and short history.....	25
2.2.2 Social Capital and Technology	28
2.2.3 CSCW example in Social Capital	28
2.2.4 Relevant Themes in Social Capital.....	29
2.3 Theoretical Limitations	30
2.3.1 Choice of theories.....	31
3 User Involvement and System Acceptance.....	33
3.1 Introduction	33
3.2 Problems with Acceptance: The Eight Challenges of Groupware	33
3.2.1 A Disparity: Work vs. Benefit.....	35
3.2.2 Critical Mass and Prisoner's Dilemma Problem	36
3.2.3 Social, Political and Motivational Factors.....	37
3.2.4 Exception Handling in Workgroups	38
3.2.5 Designing for Infrequently Used Features.....	39
3.2.6 The Difficulty of Evaluation	40

3.2.7	The Breakdown of Intuitive Decision Making.....	42
3.2.8	Managing Acceptance: A New Challenge.....	43
3.2.9	Challenges from e-mail.....	44
3.2.10	Challenges in groupware	45
3.3	Have there been improvements?	46
3.4	Workarounds	47
3.5	Management involvement and Awareness.....	48
3.6	One More Challenge: Users vs. Management.....	49
4	Methodology	52
4.1	My use of Ethnography.....	56
4.2	Possible Problems	57
5	Case	58
5.1	MOSCITO	58
5.1.1	The MOSCITO Project and Social Capital.....	60
5.1.2	What to be done – Tasks and Methods.....	61
5.1.3	My involvement in MOSCITO	62
5.2	Electric Company	63
5.2.1	Case Introduction	64
5.2.2	System Overview	65
5.2.3	Groupware Functionality.....	68
5.3	Interviews: Sampling, Motivation and Limitations.....	69
5.3.1	Sampling	69
5.3.2	Motivation	70
5.3.3	Limitations	70
5.3.4	Progress plan	71
6	Findings	73
6.1	Observation.....	73
6.2	Management Level	75
6.2.1	System Acceptance.....	76
6.2.2	Workarounds and non-technical solutions	81
6.2.3	User involvement and Management Involvement	81
6.3	Users	82

6.3.1	System Acceptance.....	83
6.3.2	Workarounds	87
6.3.3	Awareness.....	88
7	Discussion	91
7.1	Grudin's challenges, CSCW and the collected data	91
7.1.1	System Acceptance in the data	91
7.1.2	Social Capital, Technology and System Acceptance.....	95
7.2	Workarounds	97
7.2.1	Workarounds and CSCW	97
7.2.2	Workarounds and Social Capital.....	100
7.3	Awareness and Management Involvement.....	101
7.3.1	Awareness, Management Involvement and CSCW.....	101
7.3.2	Awareness, Management Involvement and Social Capital.....	104
7.3.3	Users vs. Management	105
8	Conclusion	106
8.1	Summing up	106
8.2	System Acceptance	106
8.3	Workarounds	107
8.4	Awareness and Management Involvement.....	108
8.5	Answering the Question: User Involvement and its effect on System Acceptance	109
8.6	Further work	109
	Bibliography.....	111

1 Introduction

1.1 Motivation

I have always been interested in the use of computers. I am extremely fascinated by how rapid things are developing and how soon artifacts and gadgets could go from being new and fancy to being outdated. The reasons why we choose a certain technology or software could be many faceted; political, usability, force, availability, ignorance of options or preferred taste. And it is related to how we use technology.

Whatever our choices are, we still approach the technology as individual users. Computer novices getting “assembly line” instructions on how to do the work required will still do it their own way. The output, result and steps in the process etc may be correct step-by-step progress, but the actual use will still be individual. My opinion is that every computer user has an individual way of using a computer, either conscious or not. In previous work experience

I have often been annoyed at the inflexibility of the software I have been required to use. I have felt that some functions are very irrational according to the way I work and actually slow down my routines. As I am accustomed to computers the result could be me using my own ways to get things done. And I could observe others with their own opinions, likes and dislikes of the various systems creating their own routines to get the job done.

This has led me to focusing on user involvement in development, and the thought of users knowing best what they want. But with this thought comes the acknowledgement of users being individual and not one well defined entity.

These thoughts and observations have lead me into wanting to investigate further why we accept some software and also the social sides of cooperation through technology with focus on user involvement for system acceptance.

1.2 Research Question

My research question is based within the effects of user involvement on how the system is accepted by its users. The everyday user of a system needs to be taken seriously in both the planning and introduction, but also by being generally informed and having awareness to

avoid black-boxing. One-sidedness must be avoided, and a negotiation between bottom-up and top-down should take place. The system should have both accountability for the management and take its users seriously.

My question is thus:

Does an increased focus on user involvement in developing groupware have a positive effect on system acceptance or is it just a dead end diversion?

My main focus of attention is on a case study of a groupware system involving users with different work tasks and how they are facilitated to meet the demands of a groupware application. In relation to the MOSCITO project, the groundwork for sharing social capital is not only infrastructure and software, but also the users and how they use the system. An information sharing system is nothing without the users' contributions. There can be no utilization of social capital if the willingness and knowledge of the users are lacking.

1.3 Problem Area

The complexity of having many users is one of the largest problems in groupware development. As will be explained in more detail later, the differences between single user applications and groupware are not only in technical challenges, but also the community which will use it. With groupware the human factors are more pressing. Issues like group dynamics, individual differences or internal hierarchy are of great importance when planning and designing groupware. Analyzing groupware and group behavior is difficult, but necessary for developing relevant groupware.

There are huge amounts of trust placed in large corporate software systems to control and support everyday practices, to help with organization, cooperation, efficiency etc. These make demanding requirements of quality placed on software. These requirements are extended to those who use the software. The software could be ever so good, but would fail to be useful if not used correctly. It is here the human factors enter the stage. How does one

make larger number of individuals with different skills, likes and dislikes use a common piece of software/groupware correctly?

In all the cases I have studied while preparing for the thesis, some points appear to reoccur often. The general users of groupware applications have more often than not been overrun by management decisions (Grudin 1988; 1994a; 1994b), success is determined by igniting user participation (McNally, et al. 2005), and there is a huge possibility of sabotage, misuse and indifference (Soroka and Rafaeli 2006; Huijboorn 2004; Grudin 1994a).

Some organizational software has functions that the workers need in their everyday job tasks, other functions that could be viewed as unnecessary and be neglected. Both these varieties could suffer if they go on accord with those who use them. It will be interesting to investigate what effect user participation and involvement has on this.

On the other hand there is company goals and accountability. The systems are introduced for a reason, and while it's important to make users comfortable with them they should give a return of investment to the company. This is an interesting potential conflict to consider. It seems easy to take a user stance while reading some of the articles within Computer Supported Cooperative Work (CSCW).

Groupware functionality depends on user participation to be relevant. Some social capital advocates (Putnam 2000) gives examples of "successful communities" by describing a circle of goodwill that benefits all to some degree in the end. By contributing to a society you feel that you are a part of a bigger entity, while the feeling of receiving something in return the community sense blossoms and the chance of you contributing more is increasing. This is in stark contrast to some reports of worker sabotage and general mistrust between groups we see in some organizations (Grudin 1988, 89).

During my preparation for this thesis I read Jonathan Grudin's article "Groupware and Social Dynamics: Eight Challenges for Developers" (Grudin 1994b). He lists, as the title implies, challenges that groupware developers are facing, and at that time had mostly been ignoring. These challenges were mostly in relation to the groupware users being taken into consideration, social complexities within the workplace and a top-down view of thinking. In

short that there are challenges groupware faces to make a system relevant and gain system acceptance.

I found the ideas interesting to the degree of wanting to know if there was any relevance in them. Being published in 1994 and with the rapid evolution of computers and computer systems since, I wondered if there had been any improvements in the field of groupware. I want to see if the paradigms of system development have changed into a more user friendly approach. And to find out if this has had any effect on the system acceptance of the regular users, or the systems as a whole.

I will therefore look at a current groupware system and see if Grudin's ideas have been used. And if so, what effects an increased focus on user involvement has had on system relevance and acceptance. I will use a case study performed by myself. I have decided to look at this case study individually compared to Grudin's thoughts on user involvement, as each groupware system should be viewed on its own terms (Ciborra 1996, 194-195). The following issues are of interest:

- User involvement in the planning, design, testing and maintenance phase
- Introduction strategies and implementation
- Training strategies for new and existing users
- Drifting
- Awareness and social dynamics

This will be used to describe three central themes

- **System Acceptance**
- **Workarounds and non-technical solutions**
- **Awareness and Management Involvement**

It all sums up to my definition of "User Involvement", which is not only limited to analyzing user needs and designing system requirements accordingly. It is also concerned with how involved the user feels with the system as a whole. This involves the previously specified points, and underlines that for example awareness (which is described as knowledge of what your coworkers are doing, and what one action one place leads to another place in the chain) is involved in the "User Involvement" term.

It is also important not to be too hung up on the “softer” sides of CSCW, which takes the side of the users. I will also involve concepts like system accountability; that a system has its purpose and that company credibility, system trust and other requirements may lead to some sacrifices from the user perspective.

1.4 Background

Why we choose something has always been interesting to me. Just look at how the BetaMax lost to VHS in the late 70s/early 80s, despite having a head start, being the market leader in recording equipment and being a better technology. Or how despite a general “agreement” that there are better, cheaper and more stable Operation Systems options than Microsoft Windows, a vast majority use it. The answers are not given; it could of course be because we go with the flow or are blind to the options. It could also be because the value of a system increases with the number of other users. And of course when something is easily available and well supported, it’s more likely for a novice or a person taking its first steps into a new technology to choose this.

Further on, whatever our choices are, we still approach the technology as individual users. Computer novices getting “assembly line” instructions on how to do the work required will still do it their own way. The output, result and steps in the process etc may be correct step-by-step progress, but the actual use will still be individual. My opinion is that every computer user has an individual way of using a computer, either conscious or not. In previous work experience I have often been annoyed at the inflexibility of the software I have been required to use. I have felt that some functions are very irrational according to the way I work and actually slow down my routines. As I am accustomed to computers the result could be me using my own ways to get things done. And I could observe others with their own opinions, likes and dislikes of the various systems creating their own routines to get the job done.

I performed a study in a course at the University of Oslo on individual differences and difficulties in learning how to use computers. In this study I vaguely touched the subject of company training strategies (the study was performed at schools). I have carried with me

this way of thinking about individual differences and often thought about it when I have started in new jobs.

With division of labor, specialized tasks and segmented departments there could sometimes be confusions and misunderstandings about the different steps in the chain. In some cases I have seen functions and instructions been neglected because the users don't see the necessity or understand why they should do certain things. This could also result in (or be because of) poor communication between individuals and departments. Creating an "Us & them" feeling, where one department feels that other departments are doing everything wrong and making work for them even harder. The other department of course thinks the same about them in reverse.

These situations are descriptions I have either experienced or heard about in discussions about the subject. Originally they were just thoughts and annoyances one tells after a hard day's work. But in the initial phases of my thesis these thoughts made a basis for what would later be my final approach to the subject. Some issues are best dealt with within organizational psychology or other theories/practices for creating a better work environment and improving communication. But technology has become a more and more incorporated part of organizations and corporations, and a good technological infrastructure could be a building stone for the human factors as well.

I believe that the humans and technology are not that separated. Each one influences the other in a co-dependent relationship. We have become dependent upon our technology, and the technology can't work without us. As a result of this, the human-computer relationship is a more important part than before in the structure and planning within an organization. In creating systems to support interaction between different groups and individuals, there's always a chance of the system failing because of human factors and failing to see the intricacy of humans working in groups.

As I later read in for example Grudin (1988, 1994a, 1994b) there has historically been a division between social sciences and technological sciences. While there are interdisciplinary fields like CSCW, Participatory Design and ANT, how much exchange of knowledge and experience there is in reality, I don't know. A growing trend of more cooperation between

different fields of research is a very positive thing, but I suspect there is as much discord as agreement between the involved parties.

Even personal experience from HCI courses based in both psychology and informatics show some discrepancy, and some prejudice towards each other.

My interests are also within the field of social studies, and thus I want to keep a certain degree of focus upon the humane sides of groupware development. It will not be a very technically oriented thesis. My main concern for this thesis is about working in groups and the complexity involved when different people are joined together with common goals, common software and different personalities and skills.

Human factors could perhaps be seen as something less complex than they should be by engineers and management. In creating systems to support interaction between different groups and individuals, there's always a chance of the system failing because of human factors and failing to see the intricacy of humans working in groups. The same caution that must be shown in dealing with groups of individuals in real life should be adapted to groupware thinking. Even if the cooperation and communication is done through technology, the complexities of ordinary societies apply.

1.5 Synopsis

I will start by going through the theories I will use as backing for my research. Then I will continue with an introduction to the central themes in Jonathan Grudin's "Eight Challenges for Developers" article and some other relevant views, which I will use as a foundation for my analysis. I will give an introduction to my case study and the MOSCITO Project, in addition to the methodology of my study. My findings will first be presented structurally and later discussed in comparison with my theoretical backing.

Background Theory:

In this chapter I will explain and give a brief history of CSCW and social capital. I will also introduce some central definitions and explain my choice of theories.

User Involvement and System Acceptance:

In this chapter I will give an introduction to the central article of my thesis, and list some of the central concepts I will use according to CSCW in this thesis.

Methodology:

Here I will give the theories and methodologies of my field study, along with an explanation of why I have chosen this methodology.

Case:

I will provide an introduction to the MOSCITO project along with the introduction to my case study.

Findings:

Then I will provide the data I found during my case study, divided into observations, interviews with management and interviews with users.

Discussion:

In this chapter I will compare my findings to the theory and central themes of CSCW and social capital. The discussion will be divided into System Acceptance, Workarounds, and Awareness and Management Involvement with Grudin's eight challenges divided amongst them.

Conclusion:

I will finish by telling what I got from the discussion and case study in accordance with my research question. I will also have suggestions about further work in the subject.

2 Background Theory

2.1 CSCW

I will present an overview of the history and central definitions of Computer Supported Cooperative Work.

2.1.1 History

The history of Computer Supported Cooperative Work, CSCW, began in the mid-eighties. Twenty people from different research fields gathered at a workshop organized by Iren Greif of MIT and Paul Cashman of Digital Equipment Corporation in 1984. They were all there with an interest of looking at *how people work* by exploring technology's role in the work environment. The general idea of technological group support was not new, but earlier attempts had lost momentum and faded. The previous approach of Office Automation (OA) had experienced that the main challenge of groupware was not technical. On the contrary, single-user systems had at the time (60s-70s) handled complicated technical tasks like airplane seat reservation and office payrolls, and was able to handle the demands of the time.

With the introduction of mini-computers they saw a promise of more interactivity and cooperation. The problem with OA was *system requirements*, or defining the precise requirement of groupware. They needed to know more about how people worked in groups and how the technology affected them. A point that had been promoted by notable engineers and people involved with management information systems (MIS), but largely ignored by the designers and developers actually developing group support applications (Grudin 1994a, 19).

The CSCW people realized that the main expertise in work relation, businesses and group work lay with other fields of research. They invited economists, social psychologists, anthropologists, organizational theorists, educators and others to teach them about group activity (Ibid, 19-20). They felt the existing software development directions did not suit the new field of groupware, as the requirements should be more based on the groups than on the technology. The spread of personal computers had created a large market for single user applications and even research like Computer Human Interaction (CHI) and the human factors (HF) in computing (which were more based on single user analysis).

In 1986 the first larger CSCW conference to gather more widespread support was held in Austin, Texas. The more than 300 delegates were told that the role of computers in society needed to be refocused more towards cooperative group work. The technologies and ideas of CSCW and groupware already existed in some form, what was new was the gathering of more research fields (Wilson 1991, 6).

There were several conditions that occurred in the mid-eighties that encouraged CSCW.

- Computers were less expensive
- There were technological infrastructures supporting communication and coordination
- More familiarity with computers and working with computers made workers more willing to try group software
- The quality of single user applications pushed developers to seek new ways of enhancing or differentiating products

(Grudin 1994b, 94).

With the increasing numbers of personal computers, and companies affording more than a few of these, there was a trend of networking these together. With networking capabilities it was natural to try to make cooperation tools. Lots of interest groups were intrigued with this development. Network, computer and software developers all saw a challenge in this. Academics identified the importance of usable system based on in-depth human understanding, and user organizations were interested on grounds of the dependence of team interworks (Wilson 1991, 4).

Beside the CSCW field, several interest groups were focused on looking further into the emerging fields. Amongst these were the UK government computer agency, the CCTA, and the European technology interest organization CSC.

Thus a movement to develop and release cooperation tools with a steadily increasing number of software (Ibid) with more or less success (Grudin 1988) was born. And a research field to look further into it was established.

2.1.2 Definition

“CSCW [is] a generic term, which combines the understanding of the way people work in groups with the enabling technologies of computer networking, and associated hardware, software, services and techniques.”

(Wilson 1991, 1)

CSCW has two main components: The group working process and the technology enabling it. In addition there are the business needs and demands.

2.1.2.1 Group Working Process

Groups are made up of individuals, and in order to understand how groups work one must also understand the individual's participation and contribution. Any group will have some degree of negotiation between participants to overcome problems. The group dynamics are very important. Developers must understand the workings within groups to design the appropriate tools. But the groups also work within an organizational context which sets rules, limitations and demands on the group behavior (Wilson 1991, 8).

Joseph McGrath shows an interesting table in his “Groups: Interaction and Performance” (McGrath 1984) which divides into Production (the company), Group Well-Being (groups within the company) and Member Support (the individuals), see Figure 1. The overall goal is Performance for the company, which includes interaction within/between groups and participation from the individual. Before this is obtained there are several stages with different demands from the three different categories: Inception, Problem-Solving and Conflict Resolution. I will not go further into detail of this table, but rather use it to prove the point about negotiation and cooperation between different factions within a company. It is a good model to show how the needs of both individuals and groups influence the performance of the company.

	Production	Group Well-Being	Member Support
Inception	Production demand and opportunity	Interaction demand and opportunity	Inclusion demand and opportunity
Problem-Solving	Technical problem-solving	Role network definition	Position and status attainments
Conflict Resolution	Policy resolution	Power and payoff distribution	Contribution and payoff distribution
Execution	Performance	Interaction	Participation

Figure 1 Group Functions (McGrath 1984)

2.1.2.2 Technology

The rapid development of technology provides the developers with great opportunities. There is no indication that this will slow down any time soon. As we have seen, the networking capabilities and affordability of computer systems were in place in the early years (Wilson 1991, 8). The evolution of technology has increased massively; to the point where technology should not be standing in the way of developing good CSCW systems.

2.1.2.3 Business demands

There are lots of needs in the business world, and the trends in the 1990s showed more attention to customer needs, response time, reduced overheads and the ability to gain new and maintain old markets. Computer systems are seen as a good tool to renew and build the organizations. Especially in large companies there are demands to improve communication, quality of decisions, efficiency in teamwork and organizational structures.

There is a belief within the CSCW field that these improvements could be reached, or at least helped, by good groupware systems. A challenge is to provide success and best practice software to inspire more initiatives in CSCW (Wilson 1991, 9).

2.1.3 Complexities

With the emergence of appropriate technological platforms in the mid-eighties it was plain that the social issues were perhaps more present than many developers thought. Over the first years vast investments were injected into CSCW projects, but with few results and limited monetary return. In an article from 1988, Jonathan Grudin analyzed why these projects failed (Grudin 1988). His three main factors of failure were all based on human factors.

1. *"A factor contributing to the application's failure is the disparity between those who will benefit from an application and those who must do additional work to support it."*
2. *"A factor contributing to the decision-making failure that leads to ill-fated development efforts is the unique lack of management intuition for CSCW applications."*
3. *"A factor contributing to the failure to learn from experience is the extreme difficulty of evaluating these applications."*

(Grudin 1988, 86)

These factors sum up both the initial ideas and the persevering failures of not understanding the complexities of group work.

CSCW looks at computer mediated cooperation with strong emphasis on the complexity of the interaction between humans and computers, and between humans. The focus is on the social and organizational aspects in using technology for cooperation. Because of this in-between approach the field has attracted researchers interested in both the social and technological sides of development (Grudin 1994a).

As evident by the early analysis presented by Grudin, this cooperation is important, but is perhaps not at a satisfactory level yet. It has been added here to set the scene of some of the main problems within CSCW and as a reminder of the social focus of the field.

2.1.4 Different types of technology

The types of technology used for CSCW are diverse. There are also differences between types of communication. The easiest ways to separate these are presented in this matrix (Figure 2):

	Same Place	Different Place
Same Time	<ul style="list-style-type: none">• Whiteboards• Meetings with cooperation tools• Brainstorming tools	<ul style="list-style-type: none">• Instant Messaging• Telephone Conferences• Video Conferences
Different Time	<ul style="list-style-type: none">• Post-its• Whiteboard• Information Screens	<ul style="list-style-type: none">• E-mail• Voicemail

Figure 2 Time and Place-matrix based on (Grudin, Computer-Supported Cooperative Work: History and Focus 1994a), some own suggestions

The demands of the CSCW applications are determined by its place in the time-place matrix, both socially and technological. Some transcend the borders while others stay in one category. Each groupware system will have its own needs and demands according to this (Ciborra 1996, 195). One term that is often used synonymously with CSCW is *groupware*. And in this thesis it will be the relevant application – or group of application – and could vary between the different categories, thus increasing complexity.

2.1.5 Groupware

A definition of groupware could be to take the word literally in regards to the similar sounding *software*; software for groups. However the term needs to be specified somewhat more precisely. It is sometimes mentioned almost as a synonym for CSCW, but groupware in itself does not imply that there is cooperative work involved. Groupware is more a

framework used in CSCW. The original definition was coined by Peter and Trudy Johnson-Lenz in the late seventies as "intentional **group** processes plus soft**ware** to support them". At its core it's more a class of applications used in small groups or corporations based on the merging of computers and information bases through communication technology (Ellis, Gibbs and Rein 1991, 39).

There is a need to specify my meaning of the term, as it will be used very often in this thesis. My use of the word will be of a system framework that supports communication and information sharing within an organization. It may also consist of several applications or factors, not all of them may be used purely for cooperation and not all technical, but with a common goal of information sharing and cooperation. I have my inspiration from the definition of *group* being a collective way of working, sharing and cooperating. And *ware* being the artifact or tool supporting the group work (Ciborra 1996, 194). The term groupware is in that way descriptive of the kinds of technological tools relevant for this thesis. I will thus include some aspects not directly integrated into the main tool in my groupware definition.

Groupware has also been used in cross-reference with Collaborative Software. The term collaborative (as in collaborative tools/software) hints of more simultaneous *right here and right now* co-actions. Cooperation on the other hand says nothing about the time and place. Both the technological and social focus will not necessarily be on instant communication but rather the possibility of information exchange in any way, time or place. This definition will not be used.

2.2 Social Capital

I will look at the history and definitions of social capital, and present some central areas and its relation to technology.

2.2.1 Definition and short history

The definition of social capital varies between being regarded as similar to civic society and trust by political scientists to *"the aggregate of the actual or potential resources which are linked to possession of a durable network of more or less institutionalized relationships of mutual acquaintance and recognition."* as coined by Pierre Bourdieu (Bourdieu 1983). Robert

D. Putnam, as one of the proponents of social capital, describes it as the collective value of all social networks and what these networks can do for each other (Putnam 1995, 66).

Social capital can roughly be understood as the goodwill that is engendered by the fabric of social relations that can be mobilized to facilitate action (Adler and Kwon 2002). Social capital has sprung out of sociological thinking and entered as diverse fields as technological innovation, political science, organizational transformation and understanding social change. One of the first known occurrences was written by West Virginia State Supervisor of Rural Schools, L. J Hannifan, in 1916. He writes:

"The individual is helpless socially if left to himself [...] If he comes into contact with his neighbor, and they with other neighbors, there will be an accumulation of social capital [...] The community as a whole will benefit by the cooperation of all its parts, while the individual will find in his associations the advantages of the help, the sympathy and the fellowship of his neighbors."

(Putnam 2000, 19)

This early occurrence captures much of the essence of social capital as we know it today, and putting it in the context of social involvement in public schools. The other occurrences happened in other fields, as the term was used by Canadian sociologists to characterize club memberships by "arriviste" (i.e. people who have risen in status, but not gained the acceptance of people already in that class) suburbanites in the 50s, to urbanite Jane Jacobs praising neighborliness in big cities in the 60s, via economist Glenn Loury's analysis of the social legacy of slavery in the 70s to social theorist Pierre Bourdieu and economist Ekkehart Schlicht to underline the social and economic resources embodied in social networks. The circle ended in the late 1980s as sociologist James S Coleman put the term on the intellectual agenda, using it to –again – highlight the social context of education (Putnam 2000, 19-20). We notice here that the term was used in diverse fields, independent from each other.

When described, social capital could sound very harsh with words as "exploit", "take advantage", and give references to abuse of power or acquaintances. And examples of social capital is seen even in as diverse organizations as KKK, fraternities, Free Masons and church congregations. The effects could in other words be common good for the organization and

worse for the society, or even few people getting the glory of the works of many. But it's important to underline that social capital should work in all directions with exchange both back and forth towards a common good.

Putnam even mentions the Oklahoma City bombings by Timothy McVeigh as examples of utilization of social capital. But at the same time underlines that the idea should be more in the lines of "Kindness of strangers" and "I'll do this for you without expecting anything specific back from you, in the confident expectation that someone else will do something for me down the road" (Putnam 2000, 21). It is important to underline that exploit in the social capital sense is meant in a good way.

There are, and have always been, people who "get things done". Social capital advocates identify the importance of a large, differentiated network of professional relations - and more important, the ability to utilize this network - as crucial in understanding why. By pulling the strings in their web of acquaintances they accomplish tasks that could be in the outskirts or well beyond their knowledge and training, thus increasing effectiveness.

As a comparison, Pierre Bourdieu talks about the three capitals: economic capital, cultural capital and social capital (Bourdieu 1983), and this grouping show the power of social capital. Economic capital is the monetary power to get ahead, cultural capital is the backing of a strong culture which you can exploit (coming from a rich tradition or a culture that is known for something gives a promise to your own product) and social capital which is the human intellectual resources. Robert D. Putnam on the other hand divides the three capitals as Human Capital (The manpower), Physical Capital (The tools) and Social Capital (The network).

Even with increasing research support, the idea of social capital didn't gain widespread acceptance until the World Bank devoted a research program to it, and Putnam publishing his article "Bowling Alone" in the mid 90s. Putnam wrote about how civic "responsibility" like voting, participating in committees, political party work, civic organizations etc, has decreased since the 1950s and 60s (Putnam 1995, 66-67). He then identifies the success of a community to the level of involvement and engagement from the population. Membership in clubs and societies, reading news and, in short, participating in the community "[...] were the hallmarks of a successful region" (Putnam 1995, 65).

The title is attributed to the fact that the total number of bowlers have increased in the same time span, but the number of people bowling in teams and tournaments have decreased. The lack of community participation decreases our network density and reach along with knowledge sharing. His first article created an interest and, of course, criticism Putnam even blames technologies as one reason for our declining interest, with television being the main foe (Putnam 1995, 74), and the Internet as a device to distract us from bonding (Putnam 2000, 410).

2.2.2 Social Capital and Technology

This is an interesting contrast, or possibly a challenge, to research being done today where technology is seen as a tool for developing and maintaining social capital. Social capital started as a more non-technical research field, but has become a part of the growing trend of implementing sociological/psychological theories into technological research. Social capital is involved (directly or indirectly) in prominent social and technological theories/research areas such as Knowledge Management. There are also areas where social capital is relevant but not mentioned, or could perhaps have been used as a describing theory, like the Open Source community.

The combination of technology and social capital is relatively new compared to the “humane” focus in the start. There are even examples of wanting to defend technology’s role in social capital (Uslaner 2000). The trend seems however not being on defending, but looking at technology’s role. There are several studies with this focus.

- Social capital through websites (Girgensohn and Lee 2002).
- Social capital in relation to ICT in the public sector (Huijboorn 2004).
- The social consequences of Internet use (Katz and Rice 2002).
- Some studies done by Huysman and Wulf about social capital and IT (Huysman and Wulf 2004; 2005).
- The MOSCITO project, which will be discussed later.

2.2.3 CSCW example in Social Capital

An article that underlines the social changes regarding technology looks at the "Effects of a Wireless On-Line Community on Social Capital" (McNally, et al. 2005). They see a decrease in social capital, resulting in, as an example, less political involvement from youths. What they

want to test is how, or if, mobile technologies could improve social capital, inspired by (amongst others) how SMS messages could lead to face to face interaction.

Student union representatives were picked as test objects and given PDA's with access to a wireless network wherever they were on campus, and set out to create an online community with access to information, discussion, overviews and schedules. Everything was done to prepare the Student Union representatives for the face to face weekly meetings. A control group without these tools was also observed.

The students with access to the online community got instant information, discussion boards, electronic schedules and updated tasks. They felt more informed, appreciated the updated information and felt they had more messages to pass along to the rest of the students. At the meetings, attendance went up during the period of the test and more people made comments during the meeting. The first meeting after the test saw a drop in attendance, but the level of discussion was still high (McNally, et al. 2005).

The results were almost overwhelmingly positive, but we have to see it with a critical eye. The test ran for only a couple of weeks, and the effect of students being told what to do and given fancy equipment could result in a short term boost. The really interesting results would be if the test ran for longer periods with the students left more to themselves. Then we would see if the system would be maintained and used in a fashion that made it useful, or if neglect would render it useless.

2.2.4 Relevant Themes in Social Capital

There are many aspects of social capital that can be used in work environments. The main point is that any work environment with more than a few employees becomes a social entity, with all the complexities involved. CSCW and groupware is concerned with companies of a certain size, and MOSCITO's concern is about distributed international organizations. Therefore I view even Putnam's theories on declining social capital as interesting in the context of my thesis (Putnam 1995; 2000). Problems, ideas and solutions on social issues applicable to small communities are to some degree transferable to organizations.

There are three areas within I have gathered from reading Putnam:

- Civic Engagement, which he talks about as social responsibility like voting and which strengthens societies and increases trust (Putnam 1995, 66).
- Interaction, the communication and socializing in groups which builds social capital and strengthens bonds and networks (Putnam 1995, 70).
- Belonging, which is the feeling of belonging to a community and strengthening the feelings about interaction, social responsibility and contribution (Putnam 1995, 72)

These three areas will be used as a background idea of looking at the social feeling around my case study.

2.3 Theoretical Limitations

One of my first realizations while studying computer science was that I recognized similarities between theories and the differences often seemed subtle on the surface. When going in depth there is of course crucial differences, but still the division can be confusing. If looked at from a perspective, HCI (Human Computer Interaction) for example can be explained as ideas concerning the interaction between humans and computers, with focus on making computer systems usable for humans. When looked examined closer we see that it's a divided field that has spawned many new fields with minor variations. Jonathan Grudin argues that HCI "[...] has either no home or many homes", on count of the field being "multidisciplinary without being interdisciplinary" (Grudin 2006, 56).

Meaning that HCI has many advocates, from business schools with Information Systems research, CHI of cognitive psychologists, Human Factors and engineers (and others), without these branches converging (Ibid, 54). In later years we have also seen Human Centered Computing spawn out from HCI with the difference being more focus on the practices around work than on ergonomics.

The point of these explanations is to lead up to my choices of theoretical background. With so many theories and disciplines intertwining on the same subject, one could easily get lost. I will explain my choices and briefly mention some of those I have discarded, and why. I feel that some of the theories I have discarded could have easily been used, and I have been tempted to use some ideas from these theories later on in my discussions. For the sake of consistency and concentration I have chosen to disregard some theories to concentrate fully on others. To explain my choices I therefore see it necessary to mention in short why they were chosen.

2.3.1 Choice of theories

As I have chosen to look at user acceptance in company software, or groupware systems, the field of CSCW is a logical place to start. The whole idea behind company software is using computers to reach a common goal. In a large company there are many different directions and goals, the common goal could be explained to mean the best for the company. Companies invest large amounts of money in company software to increase efficiency, but there is no guarantee for payback (Grudin 1988). Another reason why CSCW gives a good insight in groupware systems is the, in computer years, long experience and amount of research done since the beginning of the eighties. A theory is never perfect in my opinion, and always has to go through a lot of trial and error. While newer theories or new variations of older theories are necessary for evolution, a solid theoretical backing gives both theoretical weight and ideas. Sir Isaac Newton once said: "If I have seen further [than certain other men] it is by standing upon the shoulders of giants." A bit pompous perhaps, but still true. There is always someone who has had something to say about what you're interesting in. Even scientific breakthroughs are based on some other theories, not just stumbled on by chance. On that note I find that CSCW contains enough experience and debate to base my thesis upon.

Another reason is of course the fact that I am basing much of my thesis on Jonathan Grudin's "Eight Challenges" (Grudin 1994b). But looking at this alone would not satisfy my search for some of the more group dynamics related issues within CSCW. As this thesis is based within the MOSCITO project, I got an introduction to social capital. The project aims at facilitating for exploiting social capital through ICT systems. For me, this implies that the framework must be in place before social capital can be exploited. But I also felt like turning this around.

Not that social capital is an entity waiting for a framework to get it started, but that it, as a theory, can be used in getting the framework in place. The whole concept of social capital, like CSCW, revolves around groups and reaching goals. When social capital literature talks about communities, sharing resources and knowledge, belonging, trust and so on (Putnam 2000), it is tempting to try to compare this with the more technically oriented CSCW field. Here I shall fall into temptation and use social capital as a less technical counterpart to CSCW, and use it as an explanatory tool about user involvement in groupware systems.

To back up my case study I have chosen ethnography as a method. Ethnography within CSCW has a long history, and many advocates (Blomberg et al. 1993 as an example). It is for me a good way to gain experience and see firsthand how users deal with the groupware systems they are using on a daily basis.

3 User Involvement and System Acceptance

3.1 Introduction

The focus of the thesis is on user involvement and system acceptance, and I have chosen to look closer at some papers regarding these issues. My main focus will be a review of Jonathan Grudin's article "Groupware and Social Dynamics: Eight Challenges for Developers", published in 1994 (Grudin 1994b). It looks at challenges faced by developers when creating groupware systems, based on non-technical issues.

While the paper is not the most recent work on the subject, I see it as a relevant article because of its urge to developers to focus on the social complexities of groupware. This is the central theme in CSCW, and here it's structured into a list of well arranged challenges with some suggestions. In that way it is useful as a framework for looking at a current groupware system and analysing the process.

Later in this chapter I will focus on the importance of workarounds, awareness and management involvement. I will also include some views of improvements. The Grudin part will cover the needed views on system acceptance.

3.2 Problems with Acceptance: The Eight Challenges of Groupware

Even though the framework, technology and ideas of CSCW were available it had to undergo a lot of trial and error. From the mid-eighties and forward there was a great emergence of groupware in corporations to facilitate for efficiency and cooperation within companies. The cost of organizational software is high due to the fact that they are often produced in-house for internal use and thus does not necessarily have any external commercial value. The value and return of investment of this kind of software lies with increased company efficiency. At that time however groupware was a new field of software design. Many companies failed to see this risk and vast amounts of money vanished in unsuccessful projects (Grudin 1988, 85).

Many of these expensive failures were due to not understanding the unique demands this class of software imposes on both developers and users (Grudin 1994ii, 93). Even the OA group identified the system requirements as a particularly important and difficult aspect. That it is hard to understand the exact requirements of groupware because of group

dynamics, individuals and getting individuals to work in groups (Grudin 1994i, 19). There were also differences between groups inside the corporation, mainly top vs. bottom or management and workers.

The managers pulled the strings in deciding on the software while the workers had to use it. The result of this is in some cases sabotage, either consciously or not. In his 1988 article, Grudin identifies the disparity between those who will benefit from an application and those who must do additional work to support it. And that failure leading to ill-fated development efforts is because of the unique lack of management intuition for CSCW applications. These are two examples of the discord between top and bottom that was evident in the 80s (Grudin 1988, 86-87).

For a groupware application to function correctly, the workers need to put in extra effort in their daily routines without seeing benefits for themselves. As an example he uses a meeting scheduler which required all workers to constantly update their personal schedule, adding possibly a few hours every week of workload. With the result being that the manager saved some hours a week scheduling meetings. Many people needed to do extra work to simplify work for a few. With a system like this, all that is needed to “sabotage” is for a few workers not to update their schedules. Without a strict rule and consequences for not using the application, the system will fail due to the many not seeing the benefits of the few.

There are several occurrences like these, ranging from military software (Ibid, 89) to smaller corporate software. Another failure of groupware in general, is the failure to learn from experience and the extreme difficulty of evaluating CSCW applications. There had not been enough experience drawn from the many failures, but one of the reasons why, is the difficulty of evaluating exactly what went wrong and understanding the experienced gained from analyzing failures (Ibid, 87-88).

Technological research – in many cases – examines the tools and how people work with them, putting their conclusions on what the tools did or didn't do. But another approach may be needed in the case of groupware. A problem with an application does not necessarily have to be the application's technical shortcomings, but could be a result of social challenges within the workplace. The workers could see the application as another example of the management “pushing them over”, or as a threat to their job security (Ibid, 89). At the

individual level there could be a feeling of alienation from the software, or the loss of the sense of security and productivity that the old trusted and well used software provided.

To investigate this further, Grudin identified eight challenges for developers of groupware (Grudin 1994ii). These are technical challenges for software development, but are based on non-technical issues. The latter is due to the fact that there are individuals interacting with the system, which gives the interface challenges of a single user application combined with a number of new challenges arising from the direct involvement with group processes.

This is, as discussed earlier in the paper, often been identified as one of the main reasons why it is difficult to specify exact requirements, and why it is hard to evaluate in case of failure. Some of the other challenges have been expressed earlier, and is not new, but important to give a greater view of the complete picture by discussing further. One interesting part is that Grudin himself has identified some of these in earlier work, predating this article with 6 years, showing perhaps that some of the errors done in the 80s were still thriving.

Each challenge will be exemplified with e-mail as an example of success, as written by Grudin.

3.2.1 A Disparity: Work vs. Benefit

“A groupware application never provides precisely the same benefit to every group member”

(Grudin 1994ii s 96)

The fact that there's not an equal distribution of work and benefit in any groupware is something developers must have in mind. This means that no matter what, someone got to pull the weight. Groupware applications are expected to generate collective benefit, but as there are differences in work tasks and responsibility, some will have to adjust more than others. The ideal situation will be that each individual benefits to some degree, even though they don't benefit equally. The reality is not as ideal, as experience has shown that some people have to work harder to put in the additional data required, and these tools are often used by managers while not maintained by the individual contributors.

One of the means to address this problem is obviously reducing the work load of the non-beneficiaries, but as the priority is shown to be focused at pleasing the main beneficiaries this is not easy to implement. The main beneficiaries are often those who get the project funded, and thus the focus is naturally on them. However, Grudin suggests one approach which involves implementing processes for using the software that gives benefit to contributors (as a “side effect”). In the case of the meeting scheduler: a commitment for the meeting convener to act on contributions of participants.

E-mail example: With e-mail there’s an equitable balance between sender and recipient. The sender does a little more work with the typing than the reader does with the reading. But here the sender is the primary benefactor, and thus does not unwillingly lay too much extra work on the recipient.

3.2.2 Critical Mass and Prisoner’s Dilemma Problem

“Most groupware is only useful if a high percentage of group members use it.”

(Ibid, 96)

Groupware is being challenged by a number of individual users with different tastes and needs. Two people working on the same project may prefer different tools, but need to agree on which tool to use. In a larger project supported by groupware, each individual must use the same tool in a manner fitting the purpose of the tool. With reports of sabotage and reluctance to use a tool “pushed” on them, achieving a *critical mass* is essential for achieving success. However, even if the end result of a group application being the ideal that every participant benefits to some degree, the phase leading up to gaining a critical mass is dangerous. If some of the earlier adopters abandon the application at a stage before critical mass is reached, the application may never reach that mass.

In Social Psychology and Game Theory there is a theory called “Prisoner’s Dilemma”, and briefly explained it describes if everyone acts to further his or her own personal best interest, the result will be worse not only for the group but also for each individual. Or in groupware terms, if one “freeloads” on the system the system will still have some relevancy. If everyone “freeloads” on the system, the system ceases to be relevant, because no one is contributing. Adding proof to the problem in the first challenge presented: That even a degree of benefit to each participant does not guarantee success for the system.

Grudin presents the suggestion to build in motivation for using the system, perhaps making users see both benefits for themselves and for the group; encouraging them to use it and gaining a critical mass based on individual preference rather than force. Another way is reducing the work effort needed from all participants, avoiding making it a burden, and a process of use that emphasizes individual and collective benefits.

E-mail example: Critical mass can be a problem, but could also work with just one other user.

3.2.3 Social, Political and Motivational Factors

“Groupware may be resisted if it interferes with the subtle and complex social dynamics that are common to groups.”

(Ibid, 97)

The computer is a very straight forward machine, doing what it is told and confiding in a world of numbers and logic. Humans on the other hand are not driven by the same sense of logic and commands. The group dynamics of a work place may be even more subtle and complex than we think. Some of our reactions are not even conscious, but based on social conventions and our image and awareness of other people around us and instincts on how to behave towards them. In that sense a simple computer application may break the social dynamics of the work place without being aware of it.

While developing an application, the relations between persons and groups, or how different functionalities create reactions from the users, may not occur to the developers. This is simply because they are very intricate and varies between different places and situations. An example is a meeting scheduler based on priority, which emphasizes the more important meetings over the less important ones. While in technical terms, and in reality, one meeting could be more important than another, some meeting conveyors may not see it the same way and think that their meeting is just as important – or even more important – than other meetings. At the same time a meeting labelled as “less important” may fail to accumulate interest and participants.

There are other examples of systems where, if used properly, the employees would generate their own permanent records of their meanings, views and possible opposition to the

management. Not only being unethical, but also possibly creating a feeling of being monitored, mistrust between groups and even fear of job security.

Grudin's solution is to recognize how big the problem is, and leaving behind all assumptions of a rational work environment. A rational view of group dynamics excludes the "... baffling non-rational or quasi-rational behaviour individuals often exhibit", as coined by Kraemer and King (Ibid, 97). The developers need a deep and sophisticated understanding of the future users' workplace to get an idea of how things work. This could be done by working close with representative users while developing the system.

E-mail example: E-mail does have some of these challenges, but as the style could be conversational or formal e-mail can apply to existing social conventions.

3.2.4 Exception Handling in Workgroups

"Work processes can usually be described in two ways: the way things are supposed to work and the way they do work."

(Ibid, 98)

When considering the irrational human behaviour previously described, tailoring a system to handle the processes done by all the individual workers is harder than it seems. Humans improvise, create their own work patterns and display a wide range of error handling. Grudin refers to research done by Ishii and Ohkubo on a groupware system, showed that workers took shortcuts and made their own modifications to the standard procedures described in their handbook.

This shows how difficult it is to describe exact standard procedures for a work task; people create their own habits from experience and personal abilities or necessity. As there are several ways of doing tasks, the actual standard procedure is hard to determine, even if the procedure is clearly defined and stated in a handbook. It is of course easier to create a system from the available standard work descriptions, and many managers and developers fall for this temptation.

But any description of a standard procedure cannot be complete due to unexpected situations arising; situations experienced workers have learned to handle on their own due to experience, not task description. Ishii and Ohkubo even concluded that *"AI techniques*

beyond the state of the art would be required to make the system useful.” There is a need of realizing how human problem solving works and how specifications are often post hoc rationalizations of what the work is supposed to be.

One example tells of a chocolate factory where a computerized stock control and sales order processing system were introduced. The computer department reported that the workers had a production mentality of responding to problems as and when they arose, and loathed to let themselves indulge in long term planning or to adopt specific procedures. They resisted the discipline the computer imposed on them.

The response of the managers was imposing even more set arrangements in the system, believing they were working in the best interest of the company. But it created considerable problems for the ones operating the computer, and in some areas the old manual system was still used. This called for the manager to suggest sabotage and wanting to remove the system. This was overruled by upper management who instead hired new staff and went through costly measures to make the system work.

Grudin admits that this complex issue will remain active as long as there is a strong interest in many organizations to support work flow management. To avoid rational myths of how things are supposed to work, greater interest must be taken in how work is actually been done. Systems must be tailored to the actual work routines, but still include a certain degree of flexibility as *“people are not conscious of detailed organizational functioning and how changes will affect other people.”* (Ibid, 99) Groupware can’t succeed if it has to count on the management urging its reluctant workers, or even to do the drastic measures of the chocolate factory.

E-mail example: *“The asynchronous, informal nature of most email makes it flexible.”*

3.2.5 Designing for Infrequently Used Features

“If ‘to a hammer, everything looks like a nail’, then to a groupware designer, every work situation calls out for communication or coordination support.”

(Ibid, 99)

Not all groupware functions will be constantly used. To design for a system to have constant communication could only make systems less effective and increase workload. It is plain that single user application would have more workload than multi user applications. When considering co-authors working on a paper, would they want to switch from their favourite word processor only to have the possibility of sending and receiving updates from the other authors now and then? And how often would it be necessary for them to be in constant contact?

The question Grudin propose is how often our work procedures require us to work solely in a groupware environment. Infrequent groupware features must not impose on more frequent single user features. If it is possible, the best solution would be to introduce groupware functionalities in already successful single user applications. This implementation could be more sensible than introducing a whole new groupware system that expects heavy use, when in reality its functions need not be used as frequently as first thought. Introducing groupware functions in a single user application may require creating awareness of the new functions and educating workers of their use. But, as this is a more soft transformation than an entirely new system, worker reluctance may not be as great.

E-mail example: *“Email is often relatively heavily used for groupware and basic use involves few features to learn and recall.”*

3.2.6 The Difficulty of Evaluation

“Task analysis, design, and evaluation are much more difficult for multi-user applications.”

(Ibid, 100)

In 1988, Grudin saw the difficulty of analysing groupware as one of the reasons why CSCW applications fail (Grudin 1988, 87). He still holds this as a valid point when he sums up his “eight challenges”, which prompts the question if there has not been much progress in the 6 years between them. As briefly mentioned earlier, the task of evaluating single user and multi user application is two very different worlds. While analyzing groupware, you must consider that the user is not an individual person or a series of individual persons. The users are groups of individuals working both as individuals and a group. Or even as different groups of individuals functioning as a bigger group.

Individual success with single-user applications, like word processors, is not determined by the backgrounds and personalities of other users. Groupware on the other hand is affected in a large scale by such factors. It must interface with a wide array of backgrounds, tastes, preferences and roles. And at the same time respond to and understand the social landscapes and group dynamics of the work place. Analyzing groupware takes days or weeks and requires observation of group interactions and day-to-day routines.

“Field observations are complicated by the number of people involved over time at each site, the variability in group composition and the range of environmental factors that affect the use of the technology.” (Grudin 1994ii, 100)

Grudin states that social psychologists and anthropologists are absent in most development environments and that human factors engineers and cognitive psychologists are slowly being accepted. In several of his articles he has asked for more cross-disciplinary learning and in the case of groupware the acceptance of these non-technical ideas in system development (Grudin 1988; 1994i; 1994ii). Determining if a project is a success or failure is easy, but identifying the reasons behind either outcome is more difficult and requires more humanistic insight. As an example, a highly motivated group could find a way to work with faulty software or a badly planned introduction could cripple good software (Grudin 1994ii, 100).

As in the case of establishing standard procedures, generalization from experience is risky. There are always exception factors, as the possibility of overlooking critical circumstances by maintaining a generalized focus. Grudin mentions a co-authoring tool where the developers spent time analyzing the interface of a tool that would never be used due to the lack of certain features and inter-compatibility. An absence of definitive studies leads to costly mistakes and software being developed in ignorance of several decades of mistakes in the same type of software. The same predictable problems surface each time. Even though meeting schedulers are regarded as the most available and least useful groupware application, they keep surfacing all the time (Ibid, 100).

Grudin’s solution to addressing the problem is short:

“Development managers must enlist the appropriate skills, provide the resources and disseminate the results.” (Ibid, 100)

Analyzing from what he has written earlier this suggests that humanistic expertise must be included in both the development process and the evaluation. As this is shown to be a process that requires extensive observation over a period of time, the resources for this must be made available. Lastly it seems they must learn from past mistakes by looking at similar projects and being watchful for traps.

E-mail example: *“Organizational costs and benefits are difficult to assess, but the heavy discretionary use by individuals is a sign of success.”*

3.2.7 The Breakdown of Intuitive Decision Making

“Decisions to develop unworkable applications are widespread.”

(ibid, 100)

As mentioned in the last challenge, many projects start in ignorance to past failures and fail to learn from previous experience. The conception of groupware projects starts with someone seeing the need for such software, and Grudin claims that these are managers acting on intuition. The problem however is that groupware decisions can't be based on intuition in the same way as single-user applications. Decision makers and developers with experience who has successfully acted on intuition before could fail miserably with groupware, as most such experience is based on single-user applications.

While deciding on a single-user application, like a word processor, the collective gain could be ensured by choosing a good product with a success history. Deciding on a groupware application spawns from seeing one's own needs. A meeting scheduler benefits the meeting conveyor, and while taking the initiative for such an application the management sees this benefit. But at the same time fail to see the intricacy of groupware and the extra work needed from others. Grudin's own experience shows managers with superb intuition failing spectacularly when it came to groupware.

The reason is that everyone is to some degree biased towards their job. We all have an idea of what will help our job, and this is also true for managers making decisions. Even a good manager with a genuine interest in the other workers fails to appreciate the complexity of

the work place when acting on intuition. This fails in several layers during the development process. The customer contact is often with customer relations departments, who are also biased. When the project is underway, feedback is required from potential users; often those who are thought to benefit the most. Instead of focusing on for example simplifying data input, attention is instead directed at the interface for the project manager.

It is not likely to find good intuition for groupware in a product development environment, as they also have most of their experience from single-user applications. Many professions involved will lack the appropriate skills to see the complete picture, which can only be gained in collaboration with other professions. The decision maker is also unlikely to see the value of an application that benefits non-managers, even with a great collective gain.

Grudin states that the recognition of this fact was a factor in the emphasis of user involvement in socio-technical and Scandinavian collective resource approaches to IS development. The involvement of regular users does of course have certain risks and obstacles, but is an important approach. Development management should realise the risks, complexities and fallibility of intuition, and gain more input from users on all levels. The result could be fewer groupware projects, but possibly more realistic design goals.

E-mail example: Not all e-mail applications have succeeded, and trial and error has improved intuition.

3.2.8 Managing Acceptance: A New Challenge

"Much research in organizational information systems (IS) has addressed system acceptance."

(ibid, 101)

System acceptance is vital for any new system introduced. A system used in a day-to-day situation benefits from being accepted by the ones that use them. But developers are often shielded from the actual masses of users by diverse departments, like marketing, customer support, training developers and documentation developers. The customers are also adding to the shield through the use of consultants. Groupware can be sensitive to different aspects of its introduction. And contrary to single-user applications where a low start could gain

acceptance over time, groupware introduction can be foiled by only a small number of non-accepting users.

Groupware must be introduced with care and leaving nothing to chance. The developers must identify the group's problems and work processes and match their solutions to these challenges. Factors such as geographical distance and work flow could determine if the system should be introduced synchronically or asynchronous, or in a structural or non-structural fashion.

By analyzing the work place and their processes one can determine where systems are best placed, and which timing to use during introduction. Other steps are pilot groups, step-by-step introduction and giving the users a clear understanding, and even remove anxiety, of new unfamiliar features. There should also be preparations to avoid early rejection by addressing problems quickly (Ibid, 102).

Grudin believes that by adding groupware features to existing applications, the problem of system acceptance could be sidestepped. If standalone groupware is developed, the developers must design it to meet the real needs of the group members. A developer who understands the work environment should also be useful in designing strategies for acceptance too.

E-mail example: E-mail has spread more through academic and public sources than through marketing processes.

3.2.9 Challenges from e-mail

Grudin's central point is a shift from a technology perspective to a work perspective, and mentions e-mail as one of the main reasons for doing this. Developers see the key roles in e-mail as "sender" and "receiver". But in real life it could be "manager" and "subordinate". Technology does not recognize distinctions between these, but in the work place it is essential, even critical. The use and reception of software is not determined by the distinctions programmed in the software, but the distinctions in the organization.

The informal manner of e-mail has made it a success for regular workers. The possibility to send information to, in theory, anyone in an instant is what makes it fascinating.

On the other hand this informality could be a nightmare for organizational management. The formal structures and hierarchies of an organization could be bypassed by e-mail. The hierarchy do have a purpose, but research shows that 7% of messages sent inside an organization went more than one level up (ibid p. 103). This could create problems for the middle management who are left out of the process, but also the managers on top getting information and work others would get if the hierarchy worked.

The success and informality of e-mail could force organizations to rethink structures and procedures. Even though some discourage or terminate use of e-mail, the widespread use could spawn a new perspective and new challenges and those who have “outlawed” it are often convinced to go back. E-mail is thus not only a groupware success in regards to it being widely spread and used; it is also, almost on the contrary, the embodiment of some of Grudin’s warnings. It could break with internal social dynamics of a work place, or it could reinforce them. One could speculate that the reason for success is the apparent benefit of the largest group of users, while the problems are pressed on the ones who are usually beneficiaries of groupware systems.

3.2.10 Challenges in groupware

In his presentation of the eight challenges and the e-mail example, Grudin tries to shift the attention of developers towards a “Work perspective”. He asks questions if “we” as technology developers can change this perspective, or if reliance must be laid on anthropologists. He believes the whole development community to be technology oriented and driven. It is natural in a technological branch of science, but even so the problems are recognised. It’s the change of perspective that’s hard. And in the meanwhile vast sums of money are wasted on trial and error.

He sums up his solutions to the challenges:

- Add groupware functions to existing single user applications.
- Find niches where existing groupware succeeds. Example: Voice applications for travelling salesmen.
- Build on object management or shared information systems rather than those that incorporate organizational structures and work processes.
- Try to provide direct benefits for all group members.

- Educate managers and developers of the risks of groupware.
- Get a better understanding of the decision-making processes in development.

"Groupware must be more 'group friendly' than mainframe systems have been."
(Ibid, 104)

At the end he adds another request and tip: To anticipate organizational change. Technology will sometimes replace or deskill workers, groupware that handles communication and coordination can change authority structures, and decentralization can diffuse the prospects of groupware that benefits management. Like other technologies have done before, groupware could end up being used successfully in ways that were not imagined at the start of the project. By designing for flexibility and change the systems could be more relevant.

3.3 Have there been improvements?

Grudin's article raises many interesting questions, and some pointers to what could or should be done. One of the interesting questions, as the article was published in 1994, is if there has been an improvement. The most ground breaking change since the 80s and early 90s is of course the enormous success of the Internet. With it the general population has gotten more used to acting in an online community, to electronic communication and collaboration, and has also developed large amounts of social networks. Also it is not uncommon to have access to a computer at work, or even using this computer as your main work tool. It could be argued that the "readiness" for groupware applications is at a higher level now than in the early 90s.

I will however not search high and low for success stories, as the eventual success or failure is both difficult to evaluate (Grudin, Groupware and Social Dynamics: Eight Challenges for Developers 1994b, 100). A project can be deemed successful without knowing if it was pure luck or good planning that lead to the success. A groupware system is something that can evolve or drift into being successful and the essence of the means and events that lead to the eventual outcome are more important than the label "Success" (Ciborra 1996, 192).

I will instead look at a few positive examples concerning groupware to spot some important aspects in these.

There are examples of web sites and services that in some ways accomplish goals that CSCW reach for. Wikipedia builds an enormous database of knowledge by letting users contribute, message boards/forums lets users share knowledge and experiences or ask each other for help, as do newsgroups. MySpace, Facebook and other social networks could help improve social relations, improve networks and create new connections, even though the gain of this could be debated. The important thing is that they are multi user environments that have reached their popularity by a critical mass of user participation.

Even the informal meeting grounds shows some kind of "best practice" CSCW researchers perhaps should notice. What makes people use these tools of their own free will? Why are we so interested in joining communities on the net? Humans are creating communities, and have always done this. Communities are social entities, often derived from some common ground. It could be family, friends, work, common interests, neighborhoods, cities, countries, race, sex or anything (Ackermann, et al. 2004, 406-408).

It is important to note that some of the CSCW applications are not based on CSCW principles per se, but fall into this category during analysis (Wilson 1991, 9).

3.4 Workarounds

I have identified workarounds as an important factor to investigate. There is more to it than it may seem. In a deployed groupware system there could be some disturbances that could act as hindrances for "correct use". These could mean errors in the software, bad design or human factors. Even small disturbances could create a vicious circle were alternatives are steadily being more used (Ciborra 1996, 197). The alternatives could be other applications or non-technical.

Ciborra states that these disturbances could take away the "*wareness*" of groupware, replacing the technology aides with obtrusive objects. With the potential result of the users does not view the groupware system as helpful, but as something working against them (Ciborra 1996, 197). In this scenario alternatives are viewed as helpful, which is destructive for the groupware system. His impressions are that alternatives are a groupware's worst enemy. A groupware system must be learned and fed to the users.

There is a difference between drifting and evolution that applies to groupware. Drifting means slight changes in roles and functionality of the technology compared to the initial objectives and assignments (Ciborra 1996, 198). The users in the daily workflow experiences constant interventions and improvements that shape and color the system. By constantly using the system they come across new challenges and their own solutions to these (Ciborra 1996, 199).

Because of this new work practices take place all the time. These workarounds are not per se negative for the system, and are present in both successes and failures. The “plastics” of the system and the many work practices of the users meet in a learning process. More practices like these that are integrated in the system could lead to a more non-obtrusive system that becomes a more incorporated part of the work environment. Learning by doing equals more efficient integration (Ciborra 1996, 198-199).

It is important that the technology is flexible for these kinds of changes, as evident by the reports of user sabotage, workarounds and breaks in actual work practices reported in many central papers on CSCW (Bowers and Button 1995), (Grudin, Groupware and Social Dynamics: Eight Challenges for Developers 1994b), (Grudin, Why CSCW Applications Fail: Problems in the Design and Evaluation of Organizational Interfaces 1988) (Ellis, Gibbs and Rein 1991).

3.5 Management involvement and Awareness

There is a lot of talk about management involvement in some of the articles previously mentioned (Grudin, Groupware and Social Dynamics: Eight Challenges for Developers 1994b) (Grudin 1988). Often this talk is in conjunction with the failure of groupware. It is important to state that groupware systems should have a business strategy attached to it. Groupware ought to have strategic business goals.

But even if management is aware of the strategic potential of groupware this does not imply that the implementation is in accordance with these goals (Ciborra 1996, 200). This could be because of lack of intuition (Grudin, Groupware and Social Dynamics: Eight Challenges for Developers 1994b, 100-101) or Ciborra’s discussion about drifting. Implementation is a long

learning process in order to fit into the context. In the same way that drifting could form the work practices in a system it could also help fit the business context.

This process could be an improvised or radical reform. It could take time before the results are materialized, but management should be aware of this and be able to transcend the formal context (Ciborra 1996, 200). Much of the responsibilities of successful groupware lie with the management and it's intuition towards groupware, as evident from several of Grudin's papers.

In addition to this, management should be involved to assure system accountability. Meaning that the system must "answer" to someone and be able to document and deliver data that ensures its relevance (Bowers and Button 1995, 64).

Incorporated in the social factors in CSCW is awareness. Awareness of your surroundings, co-workers, coworkers' tasks in relation to your tasks, the chain of production and other factors could be useful both for the individual and the group. Awareness is best defined as having some shared knowledge and understanding about each others' activities (Dourish and Bellotti 1992).

One example is from the print industry, where the awareness was obtained by walking through the shop floor, talking about workloads, chatting and being aware of other people's workloads. The layout of the shop floor and machines provides lines of sight between users which supported ad hoc cooperation, while the built in sounds made operators hear how the work process was going. The most experienced workers could even identify small changes in the pitch of the normal sounds from the machines to know that a tray was starting to get empty (Bowers and Button 1995, 56).

When a new workflow system was introduced, it disrupted the smooth workflow causing difficulties and a need to re-organize the shop floor (Ibid, 61-63). Awareness should be regarded as an important issue

3.6 One More Challenge: Users vs. Management

Whenever there is a system being developed, someone is pulling the strings and initiating the project. In corporations these decision makers are often management, on some level

from top to middle management. CSCW is to some degree concerned about how to position a project within the top-down-axis. A gap between user groups, often with management on one side and the other employees on the other is dangerous for a project's success (Grudin 1988, 87).

One example is electronic calendars. These are meant, on a large scale, to simplify scheduling meetings, and on a personal scale to be a tool for each individual's time scheduling. With systems like these we need to see who benefits and who has to work harder. In this case the manager benefits by being able to schedule meetings automatically, probably with one click of the mouse or keyboard. But in order to make this system effective, each worker has to maintain an up to date calendar at all times. And there are factors like "*free time is really never free*" (Ibid, 85).

Systems like these could in theory end up saving one manager a couple of hours a week, but costing the rest of the workers a couple of hours every week - each - for maintenance. The inability of seeing these gaps could leave the regular users feeling unrewarded and forgotten. Workplace research show signs of employees being afraid of change, that the introduction of a new system or reorganization would leave them redundant.

A worker who thinks he has been left out would not openly embrace a new system. The top-down perspective often used in system development could result in direct *sabotage* or neglect by the workers (Ibid, 87).

A system based on cooperation depends on full support from the users. Another example is of a command system used on an aircraft carrier that failed because of top-down design where the users further down in the hierarchy neglected the system because they did not see the benefits (Ibid, 89). There are tendencies of management ignoring the disadvantages of their methods, the difficulties involved in this kind of system development and the extra work to be done by others than themselves.

One of the main points within CSCW is that users should see a benefit in using the applications, if they are persuaded into seeing the benefits for the company or seeing how the application could help them in the day-to-day work for personal benefit. There are two extremes in making the benefits visible.

One is "forcing" the workers to use it. Forcing seems like a harsh word, but could be as simple as making it necessary to use in the workflow. Like an inventory system in a store that has to be used to keep track of the stock and constantly updated. Another example is a sales system where the sale can't be done without using the computer system for registration. A situation where you have to use a certain application at work ensures that you're using it, but not that you're comfortable with it.

The other side is creating motivation by having the users see the benefits of the system, as mentioned as one of Grudin's challenges (Grudin, Groupware and Social Dynamics: Eight Challenges for Developers 1994b, 96).

However, the answer is not necessarily turning everything around. There needs to be a negotiation between benefactors, company policies and office dynamics. As the example of e-mail shows, a system with a shift of balance threatens the internal structures and social dynamics of the company. While the general user has taken a liking to e-mail it can be seen as a distraction and a way round the internal structures by the management. A system that is not accepted by the management is less likely to be accepted than the other way around. The challenge is to see the potential promises of such systems.

4 Methodology

An important aspect when writing about computer systems and the people using them is to get insight in a real life system. Even if one's writing in general, the knowledge and analysis of one or more actual systems could put things in perspective or focus attention to some details. Even add new ideas and possibilities, or remove previous assumptions.

There are several degrees of involvement in a studied case. From a direct involvement with much time spent observing, to textual analysis. This degree is determined by necessity/relevancy, resources and time consumption. The study should be tailored to meet the needs of the study. At the same time it's a negotiation process with the studied part, as they must sacrifice time and effort too. Thus gaining access to a suitable environment could prove difficult and certain sacrifices could occur (Flick 2002, 54).

My choice for observation is a qualitative, ethnographic look at one example of organizational software which includes groupware functionality. I will observe every day use and get an introduction to the system myself, as well as get a look and feel of internal and external communication on a general level. This will be done during the course of my interviews. The reason why I chose the ethnographic approach is based on advices from Nardi (1997), Blomberg et al. (1993), Flick (2002) and others who argue that the point of ethnography is to see the work where it actually happens – when it happens. Blomberg et al argues that with the evolution of CSCW there was a realization that the existing methods of analyzing the needs and activities of the users, and that ethnography and its methodology of gaining access to everyday practices is well suited for CSCW (Blomberg, et al. 1993, 124).

This is important to me as my aim is to analyze an existing system and its everyday use. Mere interviews or quantitative questions without a firm methodology would not give me, in my opinion, important insight in how the work is actually done. By having an insight in the everyday practices, certain complaints/positive views will be easier to spot and understand. I will also try to get an understanding of the system functionalities, goals and how it was introduced from someone on management level and representatives involved with the development and introduction process. When trying to understand the work, I feel it is necessary to have a basic understanding of how the system works and the ideas behind it.

Blomberg et al. uses some definitions to how to *do* ethnography. The guiding principles, with relevance to my study and my own comments, are:

- Gaining access to work in its natural setting: To have access to observe the work in a natural setting, without any artificial staging or other things that could lead to a deviation from normal practice. In other words just observing a normal work place, and at the same time avoiding affecting the routines. This also implies that the observer should not have an intrusive role, even when going to the degree of participation in the practices. It would break the natural setting if your presence is influencing how people act. It is also an issue to assure that you are trusted and that the interviews are based on willingness to ensure honest answers (Flick 2002, 53)
- Holism: The belief that some things can only be understood in the everyday context they appear. How particular behavior fits into the *whole*. When observing something in its natural setting, even the small seemingly insignificant things going on in the background could be used to explain something more important. Not all aspects are smoothly connected, but they are always connected in some way, often through conflict (Nardi 1997, 162). As Grudin states, small “insignificant” social conflicts ignored in groupware can accumulate to being a big issue when implemented (Grudin 1994b, 97). Looking at small events in relation to the whole could be a good way to understand the complexities of cooperative software.
- Descriptive: That ethnographers are observers/participants who are there to describe what is going on, not what ought to be going on (Prescriptive). The ethnographer should always take the side of the observed part, speaking for them (Nardi 1997, 164-165). Having a descriptive stance would probably allow the subjects to speak unrestrained and let their views come through. Putting your own opinions in their words/mouths would refrain from the descriptive approach and overlay the side taking.
- Member’s Point of View: Understanding the world from the point of view of those studied. One can never understand completely and see the view from within the

head of others, but try as best to understand how they organize their worlds and work. The realization that each individual sees objects and tasks differently, and that things could be more than they seem, is exemplified in two figures in the Blomberg article. They both show a circle of tasks, objects and situations described in a descriptive way from both the researcher's point of view and the members' point of view (researcher through the participants). From the researcher's point of view a bookshelf is described as "partially used" and a photo copier as "Canon NP9800". From the members' point of view the same bookshelf is a "place for shared reference books; place to post reminders, notes to myself". And the photo copier is the "only copier that will handle my oversized originals" (Blomberg, et al. 1993, 128-129). This shows the importance of getting a good insight in the world of the participants and how they see it.

- Observation: As human activity is a main interest of the ethnographer, observing human activity and behavior in its natural setting is valuable. There are several reasons, like that what people say and what they do is two different things. But it is an important aspect, as it is related to manifest behavior and ideal behavior. Ideal behavior is what every "good" employer should do while manifest behavior is what is actually done. This is very relevant to CSCW, as Grudin makes a point of it while explaining one of his challenges. That work processes can be described as the way things are supposed to work, and the way they actually work (Grudin 1994b, 98). This may lead to distortion, willingly and unwillingly by the subjects.
- Observational role: This is an important issue. There are two different extremes of approaches that can be taken. One is the mere observational "fly on the wall" role where the observer tries to disturb as little as possible. The other is *Observer participation* where the observer could go as far as take an active role in the work practices, or in other words: do the actual work of the company. There are arguments for and against both, like that by only observing one does not get the necessary depth while there is a liability of being both observer and participant. The reality is often moving between the extremes to find ones place. The focus should be on achieving trust and becoming as much a part of the community as possible.

- Interviews: Observation should be coupled with interviews and/or informal discussions. The knowledge gained from observation will get further depth by discussing them with the observed subjects, and getting to ask them questions based on some common ground gained from observation. Ethnographic researchers enter the field with some assumptions, and there is always some bias towards the subject, unconscious or not. The interviews should however be held in an informal, unstructured manner. Meaning, the questions should not be fixed on one goal. But be open and allow for initiatives and input from the subject. The ethnographer's role is to steer the conversation in the right direction, but not be discouraged by not getting "the answers they needed" or by the subject talking about something else. Seemingly unimportant information could lead to interesting discoveries (Nardi 1997, 162). This does not mean that there is no need for some structure. The data load could be immense and without the intuition to guide the subject without influencing them, it could be too much to handle. And you could lose valuable information (Flick 2002, 49). There are also concerns about location and sampling.
- Ethnography has relevance to design: This is also true in relation to CSCW. The whole concept of CSCW is about understanding the complexities of people cooperating together with computers. Getting an understanding about this is not only about having an idea about how computer mediated cooperation will work. In the Blomberg et al article there is talk about linking ethnography and design, through the Participatory Design project.

(Blomberg, et al. 1993, 124-138)

I will try to use these principles as a backing for my case study. Ethnography as a field has some very interesting points regarding CSCW, groupware, user involvement etc. On one side it is a great tool to get an insight in an existing groupware system (or an evolving one), to understand problems, challenges or successes. On the other side it contains aspects, I feel, that symbolizes the heart and soul of the ideas incorporated in CSCW. CSCW was meant to understand and evolve groupware systems and comprehend the complexities involved in

cooperation through computers. Grudin mentions how evaluation is one of the most difficult aspects of a groupware system, based on the complications of having a number of involved people over a period of time in various group compositions (Grudin 1994b, 100).

Ethnography has been used in several CSCW and groupware studies and is thus a tried and true form of analyzing my case (Bowers og Button 1995; 1999 as examples).

4.1 My use of Ethnography

I will concentrate my study on getting a good observation of the work practices in my case study. My observation role will be strictly observational and not participatory. I will talk and be introduced to users during my observation, but restrain my role to not interfere too much.

I will perform a number of interviews with users of the system. The goal of these interviews will be to get an idea of what the users feel about the system, like advantages and disadvantages, how they work with it, awareness of functionality, if the system opens for individual flexibility and if and how they use functions deemed important by management. My samples will be people who were exposed to the introduction of the system, new employees who never used the old system and people of different access and hierarchal levels within the corporation.

In line with qualitative research my sample will consist of a small group of users and people from management. Interviews with management and designers will be both on informational level about the system and to get their view of the process. The normal users will be to get comparable data to normal users feels and reactions.

The observations and interviews will be over a small period, and will not look at changes over time. Rather a snapshot of the current state compared with their thoughts about the past and future.

4.2 Possible Problems

There are several ethical and sensitive issues to watch out for in doing research on organizational software. Ethically, an employee interview has to be anonymous. The interview subject in order to answer truthfully must know that what is said will not be reported to the management. At the same time the management, in the process of negotiating to be allowed to perform the tests, would want a report of what to be done and a report of what has been done (The final thesis will also be delivered to the management).

There is no reason to sugar coat the results in order to please management, protect the users and allow the results to be published. The study will not aim at compromising the company in any way. On the other hand, if the participants are negative about the training or something other related to my study, this will of course be in the report and final thesis. Both the subjects and the company will be anonymous. There is also the issue of company sensitive information that could accompany organizational software.

To avoid this I won't mention names of integral parts of the system and will avoid mentioning some of the data I have acquired that is handed me as part of their own internal analysis. This is probably an over-cautious measure, but I feel that my current descriptions are enough to get an understanding of the system and I keep on the safe side.

5 Case

5.1 MOSCITO

The MOSCITO (Mobilizing Social Capital in ICT-based Global Organizations) Project is a collaboration project between the University of Oslo, NTNU, University of Essex, Nijmegen School of Management, University of Michigan, Telenor R&D and Statoil (now StatoilHydro). It aims to gain knowledge and insight into how global organizations and companies could use communication technologies to exploit existing, or create new, social capital. The project is partly funded by the Norwegian Research Council.

This description is taken from the project proposal:

"The MOSCITO project will generate new empirical insights and accumulate evidence in the form of practical organizational knowledge as to how organizations operating on an international scene can use specific communication technologies to develop, maintain, combine and exploit social capital. The project will focus on three central types of technological clusters: mobile applications, web-based applications and e-mails & messaging, and investigate how each of these clusters affects the deployment of social capital in organizations."

(Moscito Project Proposal)

Three forms of technologies will be the centre of attention:

- mobile applications
- web-based applications
- e-mails & messaging

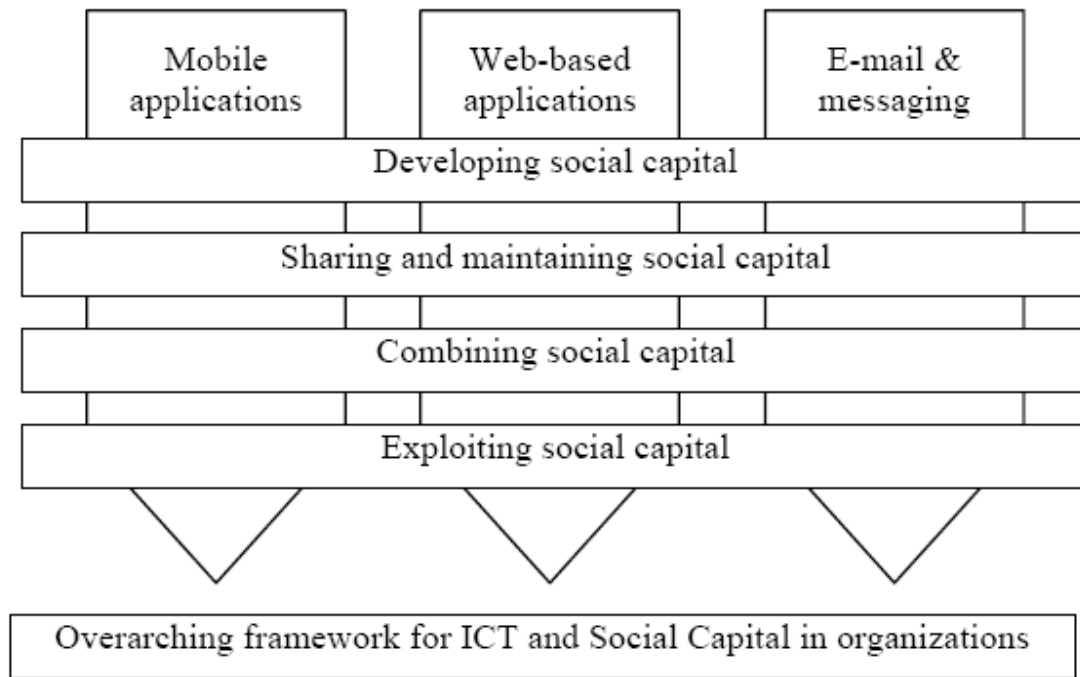


Figure 3 MOSCITO and their goal on social capital (Moscito Project Proposal)

The project will delve into how these technologies - either on their own, or together - influence the social capital of the organization, divided into four main points of interest:

1. Understand and generate knowledge as to how social capital is developed in organizations by groups or individuals.
2. Develop knowledge as to the role of social capital in knowledge sharing and in fostering cooperation between local units.
3. Achieve insights as to how and in what manners different types of social capital are combined by individuals, groups and organizations.
4. Develop knowledge as to how different types of social capital resources are exploited among groups and individuals.

(Moscito Project Proposal)

There is already widespread use of these and other technologies by themselves in the corporate world, but a need to gather knowledge of how they are utilized in relation to social capital and inter/intra-organizational networking. I will provide a look at my involvement in the project, how MOSCITO looks at social capital and their goals.

5.1.1 The MOSCITO Project and Social Capital

MOSCITO draws its definition of social capital from several sources, with the main point being access to a rich and differentiated network of relations. That social capital can roughly be understood as the goodwill that is engendered by the fabric of social relations that can be mobilized to facilitate action (Adler and Kwon 2002). Social capital has sprung out of sociological thinking and entered as diverse fields as technological innovation, political science, organizational transformation and understanding social change.

The implementation of ICT in organizations is not necessarily an evil that make employees more distant to each other, metaphorically speaking, by removing human interaction and concentrating work effort into the computer. It is more important to be seen as a challenge and more possibilities to increase and utilize social capital by the help of technological innovations like instant messaging, videoconference, mailing lists or bulletin boards. These are all tools that could help individuals initiate communication. It has been pointed out a variety of thoughts on how technology helps social capital, from the negative via the balanced to the positive.

MOSCITO argues that the social patterns of interaction both shape and are shaped by the available means of communication and that when the means of communication change, the premises for the development of social capital also changes. Meaning, it's not always gatherings at the water cooler, coffee-breaks or chatter in the corridors that generate social capital. Organizations are more and more international, and this calls for new methods of bonding and communication. Communication technologies have been a necessity in international organizations for a long time, but there is little knowledge about what opportunities social capital could unfold in a networked economy.

Drawing conclusions from ideas in other research fields show a great potential, and that social capital will grow in importance in ICT in organizations. More empirical research is needed, and MOSCITO sees gathering of information from empirical- and case studies of central new technologies as one of its goals. How ICT can be used strategically to develop and exploit social capital in networked enterprises.

There is evidence that social networks have an effect in organizations; that managers with high social capital are more productive and more likely to be promoted than those with

small networks. Through sociological studies of how communities who use their social capital are successful, it is shown that social capital is not only individual (as a group of individuals), but also important for groups. They argue that even though there is a difference between individual and collective social capital this should not be over emphasized.

Social capital is in itself a community, and is created within the community. But it can be analyzed on many levels, also the individual. The attention of technological social capital studies has been focused on the Internet and how it affects private users. As mentioned before the opinions are divided, but the technological view suggests the Internet and related technologies as means of supporting social ties. Even creating new grounds where social relations can be established.

MOSCITO has diverse Internet technologies as a focus point in its research, but asks for more research in mobile communication technologies. As a less stationary communication device than a PC, and as a more *personal* item, it has even more possibilities in maintaining social relations. The greatest social capital value being how one could be in contact with relations at all times.

What is also important for social capital in organizations is groupware, under the Computer Supported Cooperation Work (CSCW) paradigm. Unlike regular software used in organizations, the strength in CSCW lies in the fact that they should be designed for direct cooperation. Supporting and facilitating cooperation in the daily work tasks.

5.1.2 What to be done – Tasks and Methods

There are some points where MOSCITO identifies a need for improvement or change. A lot of research has been done in different fields about social capital, but MOSCITO sees a need of unifying this into a more concentrated framework, pointed at the use of ICT in corporations. There is also a need for more focus on mobile technologies, especially in empirical test studies with real users. Most of the research has been done in North America, and Europe needs to catch up and create research environments. In addition it's important to include a non-western perspective as international companies and organizations often extend into all parts of the world. There is also a need to combine different methodologies.

MOSCITO want to develop knowledge for these under-researched areas to the benefit of organizations on the international scale. The focus is social capital regarding the three technological clusters: mobile applications, web-based applications and e-mails & messaging. Through case studies, MOSCITO wants to investigate implications and opportunities for social capital. These case studies are united by a common framework covering central ideas of social capital that are of interest to highly networked organizations, including:

1. Development of social capital
2. Sharing of social capital between local units
3. Combinations of different types of social capital
4. Exploitation of social capital resources

There are work packages which aims at creating theoretical background, three cases of empirical studies (mobile, web, email), integration of the three cases and surveys of ongoing research to control the findings.

Each case study will be both qualitative and quantitative, where the qualitative approach of interviews, hypothesis and insight will be matched with a quantitative survey. A point of departure will be a social network analysis of both individuals and groups, as social capital lies within the exploitation and building of these. Traffic data of e-mails and telephones can be captured by using the information from network servers. By combining network methodology with ethnographic analysis of day to day use of ICT, they believe to get data of high quality and importance. The case studies are technological, but the project opens for comparative analysis between them.

5.1.3 My involvement in MOSCITO

I got involved with the MOSCITO project when my advisor suggested it as something that was relevant to my thesis. I got access to some internal communication spaces on the internet and got updates on the progress and future plans. I have met most of the people involved and was invited to a conference meeting between the involved parts at the University of Essex in Colchester, UK, November 2007. The case studies and general progress of the project were presented and discussed. I also presented my work in progress and got

feedback and interesting suggestions. At a later stage I was present at an update meeting at Telenor Fornebu where further progress reports and presentations were held.

My thesis and case study is not directed by MOSCITO but I was invited to place my thesis within their framework. I have drawn inspiration from their view on social capital and have included social capital theory to apply to my case study. My angle is that in order to utilize social capital through technology, a framework that is accepted by its users need to be in place. Social capital theories can be used to explain some of the prerequisites for successful exploitation of social capital.

It is valuable to have a project group to relate to when writing a thesis. The people involved have years of experience in different research projects, and have a degree of insight that makes them able to give good feedback and pointers. It is also inspiring to write a thesis to fit within this project group.

5.2 Electric Company

My case study will be performed in the customer support department of a large Norwegian electrical company. I have chosen to make the name of the company anonymous, not because of demands but as a measure of avoiding misunderstandings and possible problems. The company has been very helpful and has given their consent for me to publish this report without concealing their name. The agreement was to make the subjects names and answers anonymous, avoid revealing certain aspects of the system considered sensitive and not to use any customer information. But in dealing with such data and getting this level of access to the company, there were also some restrictions of how to use this thesis outside the masters study environment. Respecting this and considering the possible use of this thesis while applying for jobs or that the thesis is public and available to everyone who looks for it, I have therefore chosen to conceal the company name. The company will receive its own report of my findings, as they are in the process of analyzing the system for themselves and has expressed interests in any data they could get.

For future references to the company I will use the name Electric Company, or the abbreviation EC. The new system they have incorporated will be dubbed Magnolia.

5.2.1 Case Introduction

The company, EC, was brought to my attention as an interesting case. They introduced a new Customer Relation Management (CRM) system in May 2007 to replace some older systems and to introduce a more customer oriented system. Due to some acquisitions of other companies and expansions into other there were periods where customers were supported in different software according to which company they belonged to before. The new customers were gradually worked into the EC software, but the system was still a bit old fashioned. It had a focus on the measuring point instead of the customer, making it hard to focus on a complete customer portfolio. The measuring point is the connection between the main power grid and the individual customers. A measuring point focus means that all the information is connected to that specific point, regardless of who's using it.

The costumer history and logging capabilities were limited. Some forces within the company wanted a more *total customer* oriented system, a greater focus on the customer. Meaning that the goal was to gather all the available customer data in one screen, and where the old focus was on the history and changes of the measuring point, the new system would tell a more complete history of the customer, regardless of eventual address changes etc.

The new system, Magnolia, is a CRM based on Oracle's Siebel CRM. The original plan was to use the system *out of the box*, but some changes were made to keep some of the logic and terms of the predecessor. This was due to an idea of making the existing users more comfortable with the transfer. The user involvement view seems to have been present throughout the process, although it became more important as an aspect of making the transition smoother than in the initial planning phases.

In an early phase the project planning was based within a department with little or no management intervention. The problem with this however was getting the funding and necessary momentum going. After being moved to another department and spearheaded by individual driving forces, the planning and development got started. The initial idea was to get a CRM going, reducing the number of applications and increasing efficiency, and not spawned by the initiatives or wishes of ordinary users.

There were several problems underway and the system was not delivered to its original release date. It was postponed without a set release date until it was decided that it was

important to have a specific date to work towards, and to release it as it was at that date. Some of the functionalities could not be completed in time and thus the old system was still needed.

5.2.2 System Overview

As a CRM to be used in several departments by several people with different tasks, there is a lot of information to be held in one system. My focus is on the customer support department, and will describe the functionalities from that angle. The system is tailored to the specific work descriptions by regulating level of access and which applications/choices are available.

It is a web interface driven application that communicates through an API with the old system and a batch server, see figure 4. Some processes are updated instantly through the API; others have a specified time delay to allow for instant changes. Then there are some requests that are entered into a batch to be run at night when there are no workers using the system.

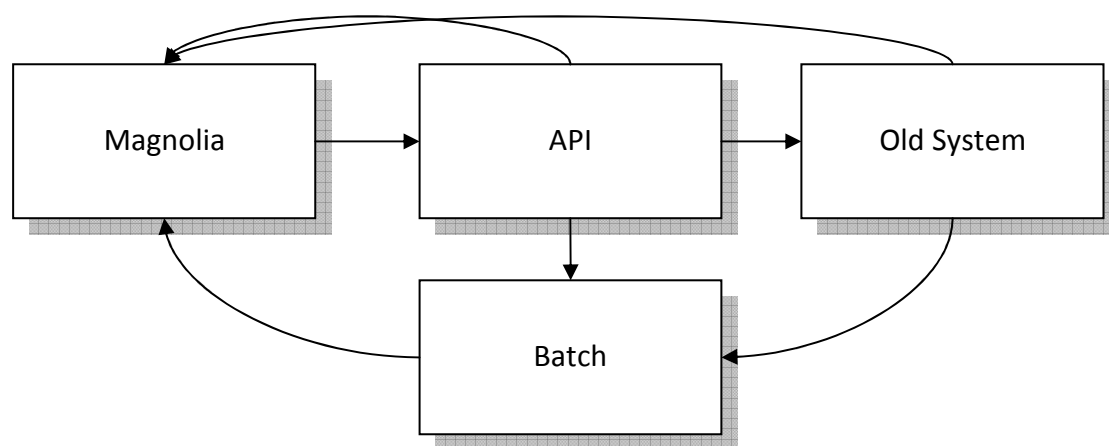


Figure 4 System update configuration

The screen is divided with a top-down, left-right logic controlled by a top menu and a search field on the right. The windows are interrelated and connected with the customer data, and have internal hierarchy structures. Leaving one window automatically saves the changes done there.

There are several types of information available, like customer information, e-mail communication, power usage, bills, measuring point information, changes, internal messages and notes, customer support cases and so on. While everything is individual information in itself, everything is linked together to form a complete picture of the customer history. While supporting a customer the users have access to every bit of information received at the same time. A log history shows what has happened, and which users have done what. Internal memos showing for example a note that the customer called about a problem, showing what was done to solve the problem, and eventual follow-ups.

These bits of information are stored in different ways. Some are updated instantly, some go to the old application which then is mirrored in Tempo, others are stored for batch processing at specific times and others have some delay before they are updated. At the present time the system is not fully implemented. The old system is still in use and mirrors information to the new system regarding invoices and the measuring point.

I will look at the most central functionalities of the system regarding my case.

A normal procedure starts with an incoming call. At a later stage, caller ID will be implemented and automatically registered in the system when the customer calls. At the present the customer consultant identifies the customer and what the call is about. The customer gives his or hers name, customer ID or other information which leads to the relevant information being found. In the system the customer consultant has a search field used to locate the customer. Some choose to punch in the phone number (displayed on the phone) if this is one registered on the customer, else a search for the name, customer ID or invoice number should help. Since the system is customer oriented, even a search on invoice number or other non-personal information would lead to the customer information being shown.

Most of the primary tasks of the consultant revolve around what has been learned in training and experience of similar cases and can sometimes be resolved without using the system. In most cases the consultant needs to find information about their customer relationship, bills or previous error reports registered on the customer. The system regulates the powers that allow the consultant to make any instant fixes on anything, but customer information (like address, phone number etc.) and certain aspects of the customer's bill.

Cases involving customer relationship, changes in subscription type, error handling, moving etc is registered and sent to the relevant department.

When registering these types of requests, the consultant creates a case. The case holds information on which customer and what is to be done, with possible additions of internal memos. The consultant registers a ToDo-function and which department it is going to be sent. The ToDo arrives at the relevant department and is delegated to another consultant with the right description and access. When the case is solved this consultant either logs and closes, or returns an Information-function with notes on what has been done, if something is ok or not. When this Information is received by the user, the situation is dealt with and the case is logged and closed.

When closing a case the consultant needs to log the event. At the time of sending the ToDo it has been logged what the call was about, which solutions were tried and that a ToDo was sent. When receiving the Information the consultant enters this into the logging, classifies the case and makes further notes on the solution or eventual problems. An overview of sending ToDo's is shown in Figure 5.

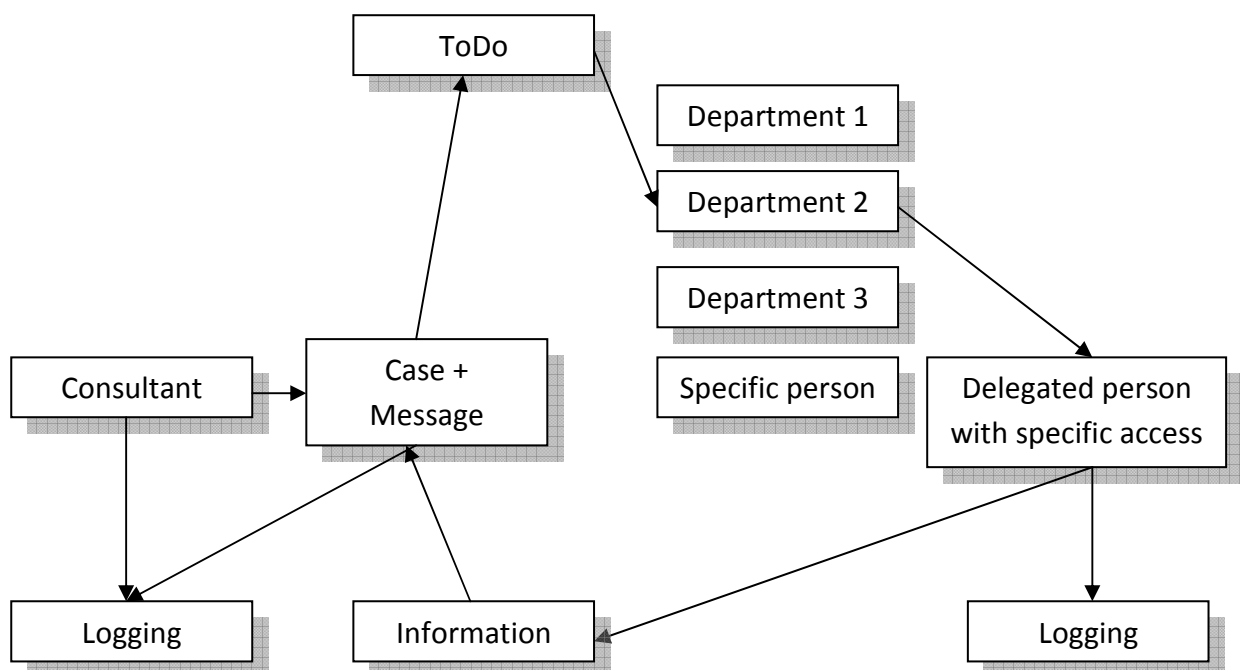


Figure 5 – Flow of ToDo's

The logging is a very central part of the system. By logging every event the customer consultants can be sure of having the complete customer history when receiving calls. The logging process has six classification categories combined with a message field to describe the cases. All the logging is saved and sorted by date and who handled the case. If a customer calls back and asks about what is happening with his/her request, the consultant sees exactly what this request was, how it was handled, who handled it and the current status.

There is also support for handling e-mail, letter and fax requests. The e-mails are entered directly into the system and connected to the right customer by the consultants. Any answers to and follow-ups from the customer will be stored in the same space with eventual e-mail attachments. When looking up a customer the consultant can view the complete e-mail history. Faxes are interpreted digitally when received and stored in Magnolia, while letters are scanned by hand before stored.

5.2.3 Groupware Functionality

The groupware functionality in itself will not be the most central issue as the pure cooperation tools are integrated into the system. The system as a whole is about information sharing and could be viewed as cooperating on customer relations (as a customer is not bound to one consultant, and each consultant logs his or hers dealings with the same customer). I will therefore look at Magnolia as a groupware system and will not separate or rule out any specific part of it. Some parts that underline the cooperation aspect, like logging and ToDo's, will be highlighted to some degree.

There are many different factors outside Magnolia that has relevance to the work being done. I will present these in the Time-Place Matrix (Figure 6):

	Same Place	Different Place
Same Time	<ul style="list-style-type: none"> • Telephone traffic information screens • Meetings • Whiteboards with messages and some presence awareness 	<ul style="list-style-type: none"> • Magnolia to some degree • Telephones • E-mail
Different Time	<ul style="list-style-type: none"> • Whiteboards • Telephone traffic information screens 	<ul style="list-style-type: none"> • E-mail • Magnolia • Intranet

Figure 6 Time and Place-matrix based on (Grudin 1994a), in EC.

Magnolia is the centerpiece but all these factors have relevance with the work being done. They are not all technical, but as groupware systems are both social and technical and have many individual factors that determine its eventual success I feel it's important to show (Ciborra 1996, 194-195).

5.3 Interviews: Sampling, Motivation and Limitations

I will present the structuring of my interviews for my case study.

5.3.1 Sampling

I ended up with interviewing two people on management level trying to get more insight in the process behind the development and introduction of the system, but also get their view on the situation. Then I interviewed five of the system users divided into three users that were there before the introduction (including one super user/team leader) and started working there at different times. The other two users had started there after Magnolia was introduced. These two were also hired at different times. The interviews were conducted as

rather informal talks structured by an interview guide that lasted for 30-60 minutes. Before and after interviews I often stayed to observe.

These in-depth interviews were combined with more informal talks with different people during my observation, in everything from listening-in-sessions on people working with customers to random talks in the hallway. I had a contact person that gave me an introduction to the work place, the people and the system. He was also available for questions and had some suggestions and thoughts about the system as well.

My observation of the system will be a snapshot of the views during my designated period. I will not follow changes but get an impression of how it is now, how it has become this way and the thoughts of the future.

I gained access to statistical data done in-house about the system introduction that are not reprinted here. However they gave me some background to what kinds of questions to ask or mental comparison data to help me analyze some of the data.

5.3.2 Motivation

I did not aim to get statistical data, but rather get a feeling of the mood and some insights in how regular users and management had experienced the process. My aim was to get the users engaged enough in the conversation to reveal the thoughts of the whole process, both the negative and the positive. My goal was not to focus on the negative sides but find things in the process that could have been dealt with in another fashion. To find data that would not only be helpful to me, but possible give the company some findings from a different angle.

For my thesis it was crucial to get this kind of case to study, and my motivation was all along to get qualitative talks with users of a system. My own experience tells me that the normal user knows more than he/she thinks about a system, and it was this knowledge I wanted to discover.

5.3.3 Limitations

I am not an experienced ethnographer. I have participated in different studies in different courses at the University, but none of this size. Experience is an important aspect of field studies, as one has to know what to look for. It is important to avoid “bad data” and don’t

miss good opportunities that present themselves. Some go to lengths in telling how bad an inexperienced ethnographer can foul a study (Nardi 1997, 363). My study is not a purely ethnographic study, but I try to employ some of their methods in the conduction and analysis of my case.

I have taken this into consideration and limited the number of participants to avoid being buried in amounts of data I couldn't handle. I wanted to have a more concentrated effort with fewer subjects.

There are some natural limitations in my case. I am in many ways an intruder in the company, as I am to minimize interruptions in the daily work. The users are interviewed in their spare time, mostly directly after they have finished working. They were participating voluntary, and we rejected those who said yes which we suspected did not want to make a bad impression by saying no.

The timeframe of the study was also of importance. The company has several busy periods where I did not want to interrupt (the users were volunteers, but my presence tied up some people to helping me arrange meetings and showing me around). The deadline of the thesis and process of getting everything in order for the case study are also limitations.

5.3.4 Progress plan

The case study had to be spread over a period of time to avoid intervening and to match the time schedules for the management subjects. Some paperwork had to be in place before things could get started. I got a general introduction and wrote my field study proposal.

After some negotiations a deal was reached in mid-December 2007. Then I got a system introduction and a chance to see it in action to prepare my inquiry. My sampling demands were used to pick which users to interview. Due to the holidays and busy periods I started my "real" observation in late January 2008 (I had done some observations during the negotiation and introduction), and started by interviewing the management subjects. There was some delay between the two because of a busy schedule, and in late February I started interviewing the regular users. My interviews were done in early April, but extended by some follow-up questioning.

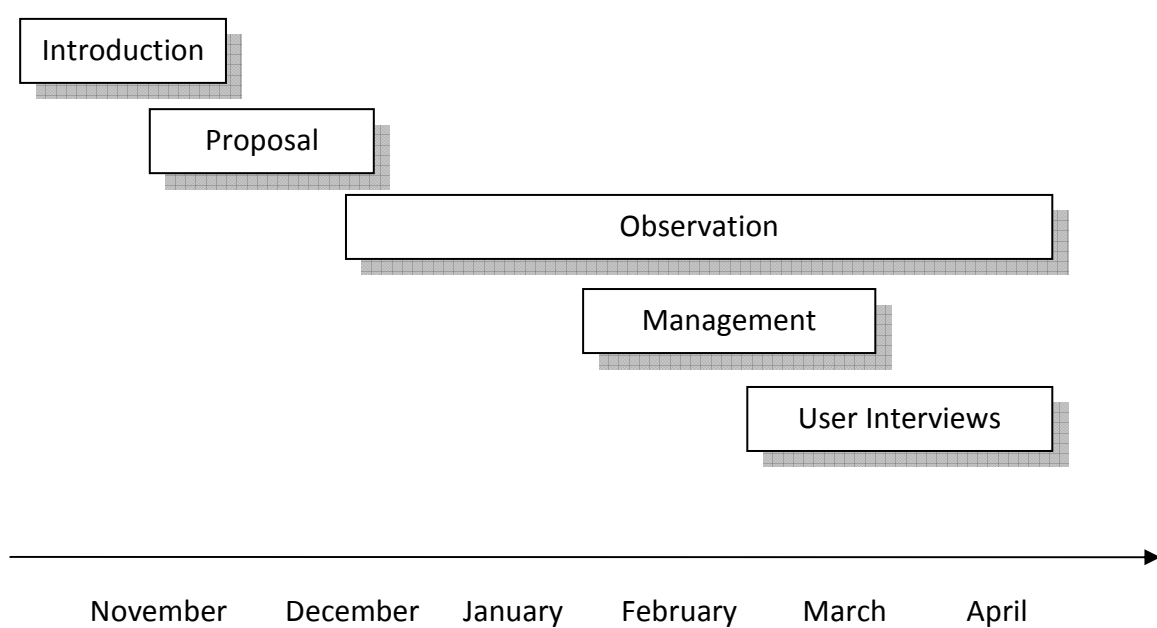


Figure 7 Timeline of my field study

6 Findings

6.1 Observation

Both before and after some of my interviews I stayed at the office and was allocated a desk I could work at in the open office environment. There were several reasons for this. It proved a good place to prepare for the interviews, and write out the notes afterwards, without the stress of getting there first. But most importantly I got to watch the work environment. At first I was a new face and attracted some attention, but after getting introduced to some of the people and hanging around for a while, this attention stopped.

My first impression was that there was much interaction between the people. They moved around and talked to each other and seemed to prefer delivering messages and asking for help by face to face interaction. Some people got asked more questions than others, and it seemed to me that everyone knew who to ask what questions. The team leaders were of course asked most of these questions, as they are the closest authority to the regular workers and have more experience. And if they didn't know the answers they knew who to forward the request to. In some cases they went to collective approaches to solve something.

An observed example was that one person encountered a problem he did not know or was unsure of the solution to. He asked the people in his closest proximity without any definitive or sure answer, and went to his team leader at the end of the desk row. The team leader seemed uncertain and did not want to give a conclusive answer without a second opinion, so these two people went to a third person. A person that the team leader meant had encountered similar problems in the past. From my viewpoint it seemed that they had a discussion that was joined by suggestions by other people nearby, so that they reached a conclusion.

I started paying more attention to the means of communication on this level, as there were functions implemented in the system for requesting support on an issue. And of course e-mail. It seemed that post-it notes were passed around frequently and that most desks had some amount of post-it notes attached to screens or laying around the keyboards. The open office space and system of on-phone/off-phone times made the room look very dynamic,

and apparently made it easier to gather up questions and messages that were not urgent and wait for a phone break to deliver them around without wasting on-phone time.

When asked about situations like these, one worker said that it was just easier to use paper or face to face interaction for these kinds of requests. Most people in the front offices are gathered in a limited space, which makes the threshold of approaching others face to face low. I got to see an example of this while listening in and observing one person while dealing with customers. If some cases required supplementary work he logged this in the case, wrote down the customer number/case number and telephone number on a post-it note, and saved it for his off-phone period. At the same time he was visited by three different persons, each with his or her own post-it notes with customer number and phone numbers written on them and was asked to call this customer (note, this was based on some customers preferring to talk to the same consultant they talked to last time they called).

When asked, he told me that he had some own workarounds to the system, based on what he felt was most effective for him. These included using the old system as a measuring point and invoice information guide located on the secondary screen (most desks have a dual screens setup). Even though it is only some cases that *require* the use of the old system, this helped him get a better overview. The new system mirrors the old system, but keeping a setup like this means you could browse through some information while having other information static and “in your face when needed”.

He had started working there before the new system was introduced and told me he was at first negative after the changeover. He felt that the training was good, but that the system felt like “too much” compared to the simplicity of the previous system. He described himself as *average +* in computer use; although my opinion after watching was that he was underestimating himself. His *average +* computer skills meant that he got the hang of the system fairly easy and that his resilience died down after a while.

The initial dislike of the system seemed like it was shared by more people. As he said it, some others in the vicinity supported him on this issue, but still there were general agreement that the new system was OK. But, as one said, it was still too unstable. It appears that unstable performance and frequent system *hangs* was included in the general view of the system. That sluggish behavior was part of the everyday use of the system, and

something that came quickly to mind when thinking about the system. They admitted it was frustrating, but saw it as a part of working with it.

Some other workarounds that were used was switching to Microsoft Outlook, composing a new mail and entering user names in the "To:"-field to get the names behind the user name. The one I observed doing this could not explain why it seemed more natural to him, but it was something he had learned once from another person working there and had continued to use ever since. Outlook is at all times open and ready, thus the operation of switching, opening a new mail and pasting the user name did not take many seconds. But it was interesting to see that there was a function for this in Magnolia that for me looked fast, easy and logical.

I also noted some spread outbursts of complaints when Magnolia was lagging. To me this seemed more matter-of-factly than based on irritation, and that they were used to this happening and knew how to deal with it.

The workers seemed in general to have a friendly and non-formal work environment, based much on humor. Everybody seemed to get involved, and the open environment provided good awareness of each other. They had whiteboards with updated information on who was having pauses at the moment, and some traffic information and general messages. A traffic information screen told them the current status of incoming calls, average waiting time and other statistics.

6.2 Management Level

Before I could talk to the regular users of the system I felt I needed some information about what strategies and plans there were. Originally I planned to only get straight facts and numbers, but realized that people involved on the project side had valuable thoughts and experience about the process. I hoped to get an informal enough conversation going to get a more realistic view of the situation than a mere presentation.

I interviewed two people with different jobs related to the planning, introduction and follow-up of the system. I was offered to interview more people with these kinds of positions, but chose to concentrate on the users due to time constraints and keeping

manageable information amounts. I will not reveal their exact positions in the company, but they had good insight and worked closely with both regular users and management in relation to the system, and thus gave me enough insight for comparison. They both were and had been involved in observations and research regarding how the users accepted the system. I was shown both surveys in progress and the results of previous inquiries on the subject, but will not refer to those here as they are in-house research. However some of the results of these surveys were in some way included in answers I got in the interviews and these will be presented here.

I will divide the answers into three areas of interest, based on my research question and experience gained during interviews. The areas are:

- System acceptance
- Workarounds and non-technical solutions
- User Involvement and Leader Involvement

6.2.1 System Acceptance

One of the most important questions is if the system has been accepted by its users and, directly affected, if it is being used as intended. I asked both my subjects their personal opinions and the straight and direct answers were “No”. The more nuanced answer was no with an added “it has gotten better”. They thought that the acceptance situation was not ideal yet, and person B thought the system had to reach a certain standard before the final stages of attaining acceptance could begin. The greatest problem for acceptance at the moment is system performance.

User acceptance was an early issue, and users were involved at different stages to obtain this. The planning and development phase did not involve users but when the system went to the test phases the everyday users were utilized. There were System Integration Tests and User Acceptance tests where the system was put to a test and bugs reported back and fixed. The Product Verification Test was to ensure that the system worked in the live environment. The users were then put through training and introduction before they had to try some tests. After the testing and training was done and the system was ready it was time for go-live. The current phase is maintenance phase where users make suggestions for improvement.

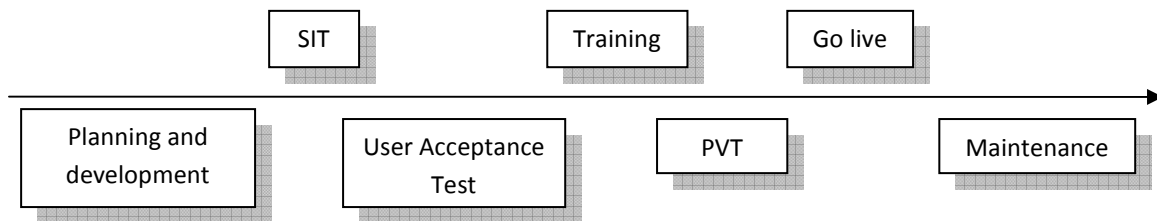


Figure 8 – Timeline of testing and introduction

After go-live the users reported that they felt the system was sluggish and unstable, and this added to the frustration of having a new system to work with. Person B told me that at the current stage the system performance of Magnolia was coming up to a satisfactory level. But as Magnolia is the interface for most users, problems in other areas of the system were blamed on it. Even if the components of the old system that was still present experienced downtime, some of these were seen through the screens in Magnolia that mirrored it and blamed on Magnolia. It seems like whenever something goes wrong people blame Magnolia.

The users also complained about Magnolia being too detailed, by requiring them to “make a lot of clicks to get somewhere”. This was seen in some way as unnecessary compared to the old system.

All this is of course negative for a system that still struggles with some initial problems. As there were several shortcomings, bugs and performance problems when the system was introduced, the strategy in the beginning was to give the users what they wanted to get them satisfied. They implemented even the smallest things reported back to them, to try and get the system acceptable as soon as possible. Suggestions of improvements and complaints of functions were to be reported to a suggestion box or through super users with extended access. The ideas are filtered and sent to cost and budget estimation and either rejected or passed on to the list of new features/corrections to be added. People involved in

maintenance and quality control also added their suggestions (often based on observations and feedback) and where also involved in the filtering and estimation (see Figure 9).

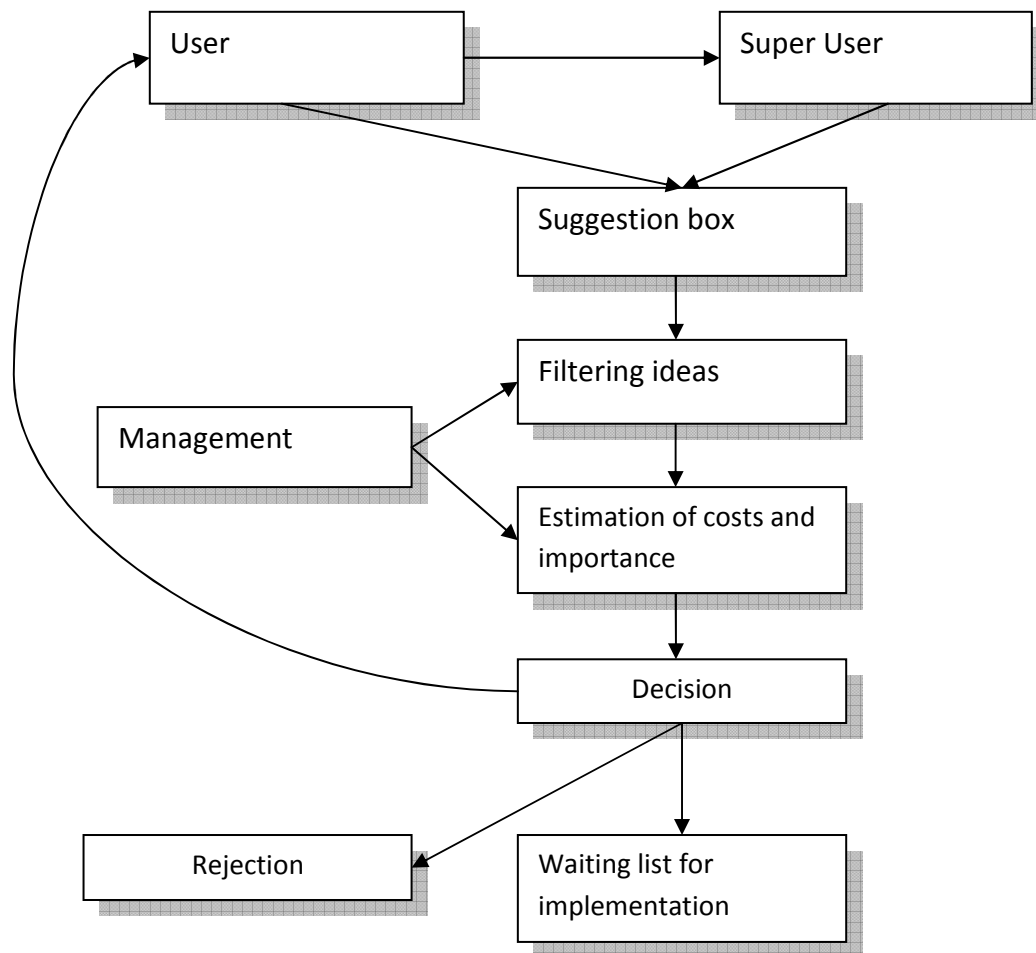


Figure 9 – Suggestion chain

Even though almost every suggestion went through in the initial phase, at this stage they are more critical of costs versus gain of the suggestions. Either way the users got feedback of the decision, and in the case of rejection why it had been rejected. Person A believed the users were very aware of this opportunity and used it well, although she meant they did not always “know what was best for them”. Not meant as negative as it may sound but there were examples of suggestions that were implemented which received more complaints afterwards.

One of these suggestions was concerning the reporting functions, as they felt the categories were too vague to determine. The solution was asking them for what categories or naming of the categories they wanted and implemented this. Now the users reported back that the categories were too specialized and therefore they used too much time trying to figure out what exactly to choose.

Both Person A and Person B thought this way of reporting suggestions was good for the system and that this level of user involvement was necessary for getting the system up to an acceptable level. They realize an importance of having a system accepted and liked by the users, and the company has done several studies based on feedback and observation about the system acceptance in the period from go live to this day.

The build-up to go-live was not perfect. Person B reported the whole process of the system introduction as quite chaotic. She felt that the users got confused when the system was delayed from its original release date. There were buildups of anticipation and promises of what was to come, and then it was delayed. This was followed by periods of uncertainty of when it was going to be released that she felt made the users “blackbox” the system. That it was something unknown that would arrive sooner or later. Person A told me that the delay was due to technical problems and not being able to provide satisfactory functionality in time. While in this period the preparation and informing of the users was upheld, but then dropped which could have contributed to a feeling of uncertainty and mistrust from the users. The date was later set for 29th of May 2007, and from that point on the plan was to keep an active information flow and a gradual buildup of awareness about Magnolia to prepare the users.

There were mandatory tryout sessions for every employee that would use the system, some were involved in testing, there were user acceptance tests in test environments and others were taught to be “super users” who would later be involved during the training and introduction. Information meetings were held and in a period before the launch it was something everybody talked about. Every employee had to go through a minimum of 3 days training before the system was launched. Those who needed more training received it, and this was planned well in advance.

However, the introduction was not as streamlined and collaborative as it perhaps could have been. There seemed to be some difference in opinion between different departments on how to accomplish the introduction process. For the individual user this had little or no difference as they only saw how it was done in their department. But for the company and the efficiency of the training it could have been more beneficiary to have a more concentrated and collective approach.

Not all of the training strategies were optimal and there were some misses in how to do the training. One example is that the test environment was quite different from working on the actual live system, and that the testing procedures did not always match what was experienced when they went live and the same things had to be done while talking to the customer and trying to be effective.

Another issue they tried to preemptively deal with was making the system somewhat familiar. Adopting some of the terms and naming from the old systems for one, but the effect of this is unknown, but adds to the measures done to soften the transition phase. They had anticipated drops in user performance and efficiency just after go-live and that the numbers would stabilize gradually. This is normal procedure when introducing a new system that needs getting used to and the goal must be to reach normal or better performance in a shortest possible time.

In short, the system acceptance after go-live was at a very low level. An unstable system, somewhat chaotic introduction, differences from training to actual use and the users missing the old familiar and trusted system was not a good start. However, Person B thinks that the future is brighter for Magnolia. In addition to the increasing stability and users getting familiar with it, they are releasing the next major update and fixes (based on user suggestions) in May 2008. She thinks the way to user acceptance is by first getting all the tinkering, detail fixing and feature implementation done and the system on an acceptable performance level. Then they should think about making the system easier, more logical and less dependent on “clicking through many pages”.

One suggestion she had herself was making Magnolia more process oriented rather than general. Instead of having to use all categories required for reporting that a customer consultant sent a price offer to a customer, they could just mark it as “Offer sent” and the

other categories would set themselves. She was surprised to discover that when asking users about this they were negative. They seemed to have gotten comfortable with the way the logging was done now.

6.2.2 Workarounds and non-technical solutions

Concerning workarounds they had not investigated this very close, but their thought was that sure it was happening. Without any concrete examples they knew that the users often had their own ways of doing stuff, besides the extended use of the old system. However, there were some top-down imposed workarounds that were due to some sub-scenarios and special circumstances not being thought of during the development.

Often these workarounds were preferred over taking investments on fixing issues, but put in the suggestion box for later. Person B thought that they needed to be fixed before the system could be considered “done”. The solution to both kinds of workarounds was concentrating on a more process oriented view based on best practices, often used functions and user feedback. User errors needed to be reported to those in charge of training to avoid similar mistakes in the future. Those who already did mistakes based on misunderstanding and misinforming should get more feedback on what they were doing.

6.2.3 User involvement and Management Involvement

During my very first interview with Person A, my thought was focused on what degree of user involvement there had been. The answers were that it had indeed been a very user oriented process. The decision to get a new system was not sparked by user wishes; neither was the decision of choice of system and the initial phases. The project got started based on seeing a need for a more complete system, and the choice of system was championed by the enthusiastic effort of one person with experience of implementing this brand of CRM's in other companies.

Still she thought there was a lack of management involvement in the process. At one time the management was oblivious to what was going on, while at the same time the user oriented view was in full bloom. This could not go on for a long time and the development responsibility was moved to another department with closer links to management.

She has an impression of the system as being *oversold* to the management in order to get things going. The promises made had not yet been met and would be difficult to meet in the near future. While both Person A and Person B are oriented towards the importance of user involvement they missed more management involvement.

The user involvement view at least gives promises of a system that is easy to use and well liked by the users. On the other hand there is a meaning behind the introduction of every computer system. While the meaning of Magnolia was defined from the start, Person B missed more management involvement in the subsequent stages. That someone higher up were asked “What do **you** want from this system?” or “What more could the company get out of this system?” rather than giving a list of what a system like this *could* do.

She did not know what would have come out of questions like that, but that it would have been interesting. And that there were possibilities of having a more concentrated approach to the development based on management and user goals. It is difficult to say in retrospect if this would have made a difference, but I got the impression that her view was that management involvement is as important as user involvement to give the system some **weight** and accountability.

This is not saying that there were no management demands, because there were. Rather an idea for getting a more streamlined introduction process and possibly adding to the system.

6.3 Users

Having interviewed someone in leader positions first I got some ideas of what I wanted to compare with the thoughts of the ordinary users. The management representatives seemed kind of glum about the user acceptance and saw the whole process as somewhat chaotic. They also meant that workarounds were necessary for dealing with the system and that not all users saw the importance of the logging. I wanted to test how the users felt about these issues.

I chose not to ask them about leader involvement in the same way I asked the management subjects because I suspected them not to have the same focus on that subject. Therefore I reformulated the section about user involvement vs. management involvement to contain

questions about their system awareness, how important they thought certain elements were for the company and how their actions affected others. This also includes issues like the training and follow-up they received.

I will divide the answers into three categories:

- System Acceptance
- Workarounds
- Awareness

6.3.1 System Acceptance

I asked every participant what they thought about Magnolia in general, to say the first thing that popped into their mind. The answers were slightly positive, ranging from “It’s a good system” to “It has its ups and downs”. They all went on to tell both what they saw was good and what they considered negative. One common denominator for the negative side was the system performance.

The old users told about the first period after go-live when the performance was horrible and everyone, including themselves felt negative towards the system. This possibly influenced their current view in both ways. Some saw the increase in performance and acceptance, while others held on to some of the negativity.

They all told about how they thought the system performance influenced their tasks. Both with how tasks took longer time to carry out and with some added frustration. The older users took this as a cue for using the old system, and to some degree it was possible that it lead to more unnecessary reliance of other tools even when the system was up and running.

In general, besides the system performance, the old and new users had some different complaints about Magnolia and the introduction that could affect system acceptance.

Old users:

- **Less accessibility:** The old system gave the necessary information up front. They felt they could read more out of fewer screens. The old system felt more tidy and the users more safe using it.

- **Less reliable:** Old system felt more robust with fewer hang ups. It was more powerful in that you felt that what you had better control.
- **More problems:** General view of there being a lot of problems in Magnolia. Not everyone could identify all the problems, but there was a feeling of “Not again!” when a problem was reported.
- **Unfinished:** Some functionality is lacking and others are illogical, all in a way that makes the user have to use the old system as a necessity.
- **Not suitable:** “Not suited for customer support, more suited for sales department”. Some felt the developers had not taken the speed requirements of customer consultants into consideration.
- **Overlaborate:** Too much to do to get little done compared to old system. Each process feels thorough.
- **Slow:** Slower response, longer processing time.
- **Less training:** 1 month training in the old system, 3 days in the Magnolia. Some users just finished their 1 month training in the old system and had not worked for long before having to learn Magnolia.

New users:

- **Problems:** Unsure if certain information is correct or not because of system problems. Feel they can’t rely on everything in the system.
- **Old system still needed:** Not confident enough in the old system if Magnolia crashes.
- **No training in old system:** They did not learn the old system before the personal training and guidance on telephone by old users.
- **Difference between training and real use:** Felt like a completely different system between the training and the actual use. At least one person felt that some of the focus during the training did not match what they experienced on the phone with customers. One example being that they did not learn logging during training.

Common:

- **Detailed:** Many information screens and a lot of clicking between them. Especially with system performance problems this was seen as negative.

There were many things that were seen as positive about the new system. All but one of the participants had an optimistic view about improvements and their own acceptance. Some were more optimistic than others, but they all felt their complaints were being listened to. Those who were not too fond of the system realized that they had gotten used to it and would continue to do so, and had no problems using it even though they thought the old system was better.

Some common praises:

- **Detailed:** You get all the information needed in one application.
- **Visual:** Less reliant on knowing keyboard shortcuts and back entrances, everything is listed in the menu. The old system was more based on keyboard commands.
- **Messaging:** Good to have a formalized and logged way of communicating between departments.
- **Logging:** In the old system there was only a message field, now there are systemized logs that show customer history.
- **Advanced:** If you know your way around the system you can find and do anything.
- **Mail:** It's got a very good system for e-mail correspondence with customers, with a systematic overview of all the mails and attachments.

In general the mood about Magnolia seemed to be on the rise. One of the most important aspects was that they thought the people responsible for the system took the challenges serious. One person told me that when she reported something about system performance she saw the anguished faces of those who were required to troubleshoot it, and thus knew that they took it serious and was eager to find a satisfying solution that would fix the problems for good.

The suggestion box for reporting errors and possible improvements seems to have been used well by the old users. Newer users did not use it as often. They told me that because

they were new they felt the older users knew more what was needed. If they saw something that they did not see as optimal, they didn't report it as they did not know what would make it better. Or they adapted to it and worked around it, feeling sure that someone else had reported it.

When asked what they thought the general feeling for the system amongst the other users was, they all answered that it was the same as they felt or possibly a little better. There was a difference here between the new and the experienced. Those who had been there for the old system remembered the first period of negativity. As the system has improved gradually, it has not been as big a water cooler-subject as it were, and thus they felt that (besides the hang-ups and occasional bad performance) people had grown accustomed to using the system.

The new employees did not have the same background and had only learned about the initial negativity from talks with other users. When asked the question of general acceptance, they told me that they were in awe of how skilled many of the new (and of course old) users were with the system. They were also impressed how quickly people adapted. They had no experience of negative mutterings amongst their cohort. There had only been some initial talks after the first lessons about how advanced and difficult it appeared to be.

I asked the users what their first impressions of Magnolia were. When I got the introduction and first observed the system I had some problems getting a grasp of things. A bit colored by this impression I wanted to know something about the accessibility of the system.

They all felt a little overwhelmed when first introduced to the system. The old users because they were used to the simple interface of the old system, the new users because it was their first meeting with the company's software and it looked advanced. But despite this first impression, the training gradually made it more understandable. The new users told me they got, to a degree, accustomed to the system fairly quickly. They gave high marks to the training staff (some of which were their future team leaders and team super users) and were impressed with how much they knew.

I got the impression that even though they did not understand everything, the training helped them get a relatively quick understanding of some of the basic functions which helped soften the impression of the system for further learning.

The last test for experienced users regarding their acceptance of Magnolia was the question of how they would deal with the situation if all (visible) traces of the old system were removed. Some answers were a bit negative, with one case (possibly humorous) believing that “all hell will break loose”. But most answers were slightly more positive. If they cleared some of the barriers and added the necessary functionality, then they would have no reservations about adapting to using only Magnolia.

At the moment they were too accustomed to having the old system on the side of Magnolia to want to depart with it. But they all knew that the original, now eventual, plan was to remove it from the front offices, and therefore knew that they would have to stop using it at a point. It is somewhat uncertain if this knowledge and increasing support of Magnolia has decreased use of the old system, but my impression was that it was still extensive.

New users were naturally more positive about how things would be if the old system were removed. Although some of them liked the simplicity of it (and one had an impression of it being more reliable), they were not confident using it and it would not be a big problem doing everything through Magnolia.

6.3.2 Workarounds

After trying some observation and getting some impressions of what kinds of workarounds were being done, I tried asking the users directly if they were aware of any workarounds themselves. Even if the question perhaps was too direct and vague, I got some examples of necessary workarounds.

One workaround was necessary to get the correct read-off from the measuring point when a customer was moving out, another was to get the. These were workarounds based on necessity, but others were based on personal preference of experience.

The older users utilized the old system for what it was worth, preferring to have it for quick references and if Magnolia was unstable. Their workarounds seemed more based on

avoiding using Magnolia. The newer users talked about completely different ways to avoid problems.

The performance problem is the largest issue and the new users I interviewed had interesting takes on dealing with this. One had two windows of Magnolia open in case one of them was slow or crashing. Her experience showed that while one of the windows was lagging or not responding, the other worked fine. Then she could continue to help the customer and eventually the first window with the requested information would come to life again with the results.

Another of the new users thought that he experienced most sluggish behavior when he was “clicking around” for information. He tried his best now to adapt to a way of working where he reduced the number of clicks. It was not always possible but to some extent he felt it helped. He had also discovered that it was two different ways to start logging a case. One of them often resulted in slow performance, while the other did not.

Most of them admitted to using post-it notes extensively, but thought it was not a backwards way of thinking. When dealing with a customer on the phone they often had to do more than one thing at a time. They could be searching or updating one type of information while at the same time the customer is giving some other information. Having a post-it note at hand and quickly noting new information felt natural, quick and easy.

The problem with asking about workarounds is that many of them gets incorporated in the routines and is not easy to remember specifically. They all at some point observe each other work. One person told me he was often astonished when he saw somebody do a little trick or twist he had not thought about. It would however not always feel natural to him to do it the same way, but when watching the other person do it his or hers way, it would always seem smarter.

6.3.3 Awareness

I wanted to test how involved the users were with both the system and their surroundings. If they their place in the chain, the consequences of their actions, if they knew the reasons behind certain actions, and so on. Some of my management findings suggested that perhaps

users lacked understanding of the importance of logging correctly, and that some users were mentally blackboxing the system.

When talking to the users about logging, they all agreed that the logging was very important. They could all see how they benefited from a detailed and well kept log, and that they could offer the customers better service this way. There were some, in regards to awareness, minor complaints about the design of the logging and how the log history was presented. But in general they all regarded the logging as one of the really good functions in Magnolia.

In regards to the management gains of logging most of them did not know exactly what the management got from the logging, but some of them guessed that it had to do with statistics about what kinds of inquiries were made. And that it could help improve in some way how the department dealt with customers.

They all thought they had got the hang of logging correctly, even though they all admitted to being unsure from time to time. Some of them thought the logging was not performed optimal by everyone yet. It was not a major issue, and encountered seldom. The view was positive and they believed it had improved and would improve even more. One interesting aspect was told me by one of the new employees, that they had not learned logging during the ordinary training; they learned it from the personal guidance sessions when they tried their skills on the phone.

The same user told me that the system and training was somewhat difficult to grasp because she knew little about the company and the interaction between departments. In short some actions were difficult to understand because the consequences and meaning of the action seemed vague. Another new user caught up on this saying he had trouble understanding why some actions were necessary until he started doing it for real. Some still had vague “Why are we doing this?” thoughts, but was unable to express/remember exactly what processes they meant.

The new user, who had felt a need for knowledge about the “complete picture”, didn’t think this was necessary to learn during training because not everyone thought that way. A more experienced user with added responsibility told me that it was this kind of inquisitive nature that had transformed her from “a Magnolia hater to a Magnolia lover”. She had been very

negative during the introduction, but when she learned the system and got more insights in how and why processes were done she thought it was brilliant.

By getting added responsibility and extended status within the system she learned more and more about it and saw the processes in a whole new way. She seemed to think that an in-depth knowledge of the processes and chains of events would tilt people more towards the positive view of Magnolia. With her positive view and added responsibilities she was aware of her role in training and seemed to want to “spread the good word”.

They seemed to be informed about much that was going on in the company and a normal morning procedure was to log in to the intranet to read the news. This was seen as a necessity to see if there was anything they had to know. This page is updated with events that could affect the demographics of customers calling. As an example, if some power lines were down in a specific area people from that area would call in larger numbers than usual, and the consultant knew exactly what to tell them right away. The page also included other company news. In addition there’s a social activities page which informs the employees on the lighter side, with non work related events.

When they encountered troubles in Magnolia, either technical or professional, they all knew who to talk to. Specific persons in each team were delegated responsibilities as super users, and the team leader was a contact person for every kind of requests. In addition, when they had worked there a while they started knowing who knew what. An awareness of certain people’s skills and experience could make the users go directly to them.

There seemed to be a very positive work environment which according to one of my subjects was especially strong within the teams. This helped in knowing who knew what and keeping a general awareness of their coworkers.

7 Discussion

7.1 Grudin's challenges, CSCW and the collected data

Jonathan Grudin states that system acceptance is vital for every new system being introduced. The system needs to be accepted by its users in order to be successful. The planning and introduction is important for managing acceptance, and it is a delicate process. A wrong turn could mean that the system is rejected at an early stage (Grudin 1994b, 101). Some points are central in his recommendations for managing acceptance, and these will be compared to the data collected from EC.

He recommends adding groupware features to existing applications to sidestep the acceptance problem (Grudin 1994b, 103). This is also valid in his challenge of designing for infrequent use (Ibid, 99). His concern throughout the paper is that groupware could be seen as added applications on top of everything else. That it is something that has to be used in addition to the "everyday" applications, and since applications like these could be used infrequently it could pose a problem of neglect.

By incorporating these features in single user applications he thinks that the acceptance problems of these features are sidestepped.

7.1.1 System Acceptance in the data

System acceptance is of vital importance to a groupware system, and I will look at how the system acceptance status in EC was compared with some of Grudin's challenges.

7.1.1.1 Managing Acceptance

Looking at Magnolia there are evidences that this kind of single user-integration holds true. The groupware features, like information between departments, logging history and so on, are not seen as an extra feature. They are incorporated in a way that makes it a natural part of the system and none of the users had any issues with it.

There are some differences from Grudin's reports that make me feel that Magnolia is treated and experienced more like a single-user application. Grudin talks about groupware being very sensible to quick acceptance, while a single-user application can allow itself more time to be accepted (Grudin 1994b, 101). In the case of Magnolia the initial system

acceptance was low. However, all users told me that they had become more and more positive as time went by and bugs got fixed.

Their issues with the groupware functions were more directed towards the general feel of the system; that there is too much “clicking around”. The problem with the logging, that they have to take some time after each customer to log the call, could be seen as extra work or that it was an extra groupware function that is somewhat intruding. Some users reported initial problems with getting the right categories, and the management was not completely satisfied with the quality of the logging. I don’t see this as an acceptance problem, because all users saw their own benefits from logging and thought they all got better and better at it.

The introduction process was not optimal. My management subjects felt that it had been a bit chaotic and that the strategy overall had failed. In the department I investigated however it seemed that they had kept a high level of information. The old users I interviewed had all heard about Magnolia before it arrived and felt that they got good information building up to the release. Some were involved in user acceptance tests; others were trained to be super users to be utilized during the training of the rest. While talking about managing acceptance, Grudin lists giving users clear understanding, removing anxiety before release and pilot groups as means of avoiding rejection. They did to some degree do this with Magnolia, but it did not remove the general skepticism. The users varied from being very skeptical to looking forward to the release, but the ones I interviewed were all somewhat disappointed after go-live.

Maybe the planning of the introduction was, as my management sources stated, faulty. Grudin talks about the importance of planning *how* the system is introduced. A good analysis of the workplace should take place to determine which strategies would be better suited. I did not gain insight in the exact plan for introduction, but my sources were themselves dissatisfied. The delays and periods of uncertainty probably did not help, and when the system was introduced it was all done with quick training and simultaneous installation.

I got a feeling that the old users did not feel completely ready when the system went live. Some training errors probably did not help the feeling of missing the old trusted system.

Grudin goes on by saying that it is a good idea to address problems quickly to avoid early rejection (Ibid, 102). This was done with Magnolia, where it was determined that the users should get what they wanted in the early phases to secure acceptance and get the system at a satisfactory level. My interviews told me that this point probably was crucial for attaining the rising system acceptance. All the old users had made use of this offer and saw some results. By adding this measure the users felt that they were taken serious.

Grudin talks about developers that must identify the work practices (Ibid, 102). The work practices in Magnolia were meant to be fitted the users of the old system. They had intended to use the system “out of the box”, but implemented some familiar terms from the old system. The users view of the work practices were overshadowed by the “clicking” problem, so it’s hard to say if there is some success in that area. Some reported that there were differences between the practices they learned during training and what they actually met when working. However, the management seems set on creating a more process driven approach based on user practices. It would be interesting to follow that further.

7.1.1.2 Critical Mass

The critical mass problem is not significant in Magnolia. Because most functions are gathered in one system, the users have to use it in their everyday work. If looked at from a slightly different angle, the fact that most old users still use the old system more than they should, could be viewed in this light. And the number of people accepting the system could be the critical mass.

Grudin presents the Prisoner’s Dilemma, with the point being that individual gain may not be the best for the group (Grudin 1994b, 96). I see some connection to the old users’ comfort with the old system compared to the new users not getting training in the old system but is urged to use it by the old users. The best for the system and the group is that the new system gains as much support as possible. As Ciborra states, alternatives are the worst enemy (Ciborra 1996, 198).

By teaching the new users “bad habits” (even though in many cases these are good advices), it could potentially end up with Magnolia lacking critical mass support. The old users are comfortable with the old system, and it is possibly a gain for them using it. However for the

group, in the future when the old system is removed, this could lead to an unnecessary drop in performance.

Grudin's answer to the problem is to build in motivation for using the system. To make sure that every user should see the benefits, for them and for the group (Grudin 1994b, 97). As I see it, the motivation is there because they all see both their own and others' benefits of the central groupware functionalities in Magnolia. The problem lies more with some of the performance and functionality issues, making especially the old users want to use the alternatives.

It is positive in some sense that the new users are not comfortable with the old system, even though some have expressed a liking towards it. In this aspect, the critical mass will continue to grow. The old users who gain more responsibility also see more and more the positive sides of Magnolia, adding to the positive trend.

7.1.1.3 Work vs. Benefit

The users did not see any personal disadvantages with using Magnolia, aside from the shortcomings that initiated some frustration. Grudin acknowledges that there will be some differences in work distribution; someone must pull the weight (Grudin 1994b, 96). It is clear that the regular users do most of the work of maintaining the customer database in Magnolia, while the management reaped the benefits.

But as the users saw benefits for themselves in doing this extra work, this is not a big issue. While the logging could be viewed as tedious extra work, it makes other parts of the job easier at the same time. A good history record of the customer enables the consultants to be more precise and possibly more professional towards the customer. Everyone I interviewed said they had no problem with the logging and saw their own benefits. Other functions as the inter-department ToDo's were also welcome.

This is in accordance to Grudin talking about "side effects". If it is not possible to equally distribute the workload, some other benefits must be given (Ibid, 96). My interview and observation data gives proof to this being a good angle, as (apart from some non-consistent use of the logging) there are no major problems or issues with workload and benefit, and the users were maintaining these functions.

7.1.2 Social Capital, Technology and System Acceptance

I have looked at Putnam's views on community and social capital. Putnam is clearly no computer scientist but his ideas about social capital are interesting (even if they are seen as somewhat controversial. (The book "Social Capital: Critical Perspectives on Community and Bowling Alone" is in my opinion a good collection of articles written against or discussing Putnam's view (McLean, Schultz and Steger 2002)). He does however take the stance of blaming technology for diminishing social capital, and we can speculate that the reason for this is a lack of technological insight.

It's interesting to see him identifying technology as an enemy of social capital, and not a helping tool as others try to argue. As an answer to Putnam's critique of the Internet, Eric M Uslaner wrote an article trying to clear it from these charges. But at the same time not hailing and over-emphasizing its importance. He argues that "The Internet neither destroys nor creates social capital" (Uslaner 2000). He views the net as more of a vehicle for mirroring everyday life, and not something that will change people. Trusting people will still be trusting, and vice versa. Thus he sees the possibility of creating communities on the Internet, but does not see the possibility of people changing personalities or that these will replace the dwindling civic communities of Putnam's golden 50s.

In recent years the involvement of social capital in technological research has become a big field, as evident by the MOSCITO project. I will not discuss any further if there is any difference between technical and non-technical views on social capital, but rather try to use the non-technical view on my collected data.

In relation to social capital and System Acceptance I have chosen to look at Civic engagement, as talked about by Putnam's "Bowling Alone".

7.1.2.1 Civic Engagement

The point is how civic engagement has created and improved societies, and Putnam even goes to the lengths of claiming that social networks as described before are not a result of socioeconomic modernization, but a precondition for it (Putnam 1995, 65). The picture painted for us is that of a small community building their community house as a gathering point for the town. Very idyllic, but he claims historical data tells us this has been a tool for creating better schools, better economics, less crime and improved government. All this

attributed to social capital. It is then important to see what Putnam sees as a key element in social capital: social trust to facilitate cooperation. With trust come cooperation and mutual benefits. If you trust someone, you can ask that someone for help - or give your help. With trust you could see a benefit in helping an individual by trusting that the favor will be returned in some way. In that way a society that *pulls the weight together*.

What I saw during my case study was a work community which seemed to be quite harmonious. I can't positively say that there are no conflicts under the surface, but the important thing here is that they seemed to have a policy of being very social. The users were in general positive to the people training them and reported of a good mood especially within the different teams. They also participated on social gatherings outside the workplace, which could prove to strengthen these feelings.

The number of Americans, who spends social evenings with their neighbors more than once a year, has decreased from 72% in 1974 to 61% in 1993. However the number of people spending social evenings with people not from their neighborhood has increased in the same period. Something Putnam thinks could reflect the growth of workplace-based social connections (Putnam 1995, 72), because this is one of the only places where interaction has increased.

At the same time as there have been a general decrease in social interactions there have been a simultaneous drop in trust. We could speculate that trust is in direct relation to civic engagement/good neighborliness (Putnam 1995, 72). But regarding my case I see evidence of people being trusting towards each other. From my interviews I gathered that the team feeling was very strong, and from my observations and talks I noticed that people took their jobs seriously and did not want to put any strain on their coworkers by acting selfish.

I have a feeling that at least part of the explanation of why the system has increased in acceptance at the rate it has could be found in this community feeling. Right after the system was introduced there was a lot of negative feelings; almost to the point of a catastrophe. At this point the community possibly dragged the system acceptance down because even people with "power" and influence were negative. But at some point this changed. At some point the system became more stable and more and more people believed in it. The most important thing was that people with influence like team leaders

and training staff conveyed the positive thinking on newer employees. As they reported trust in their superiors, a superior that advocates a positive view on a system would avoid the negative mutterings, and inscribe a positive view on the prevailing problems.

In another view, this trust could possibly have some negative sides, as the experienced users urge new employees to use the old systems and transfer some of their bad system habits based on the old system. But they were collectively learning from each other and some of those who were employed after Magnolia have reached a high level of proficiency. Thus the community is gradually improving its acceptance and skills.

7.2 Workarounds

I will discuss some of the workarounds I spotted during my case, in relevance to CSCW and social capital.

7.2.1 Workarounds and CSCW

Workarounds are seen both as a problem and something positive in CSCW (Ciborra 1996). On one hand it is something that could destroy system acceptance and make groupware systems irrelevant. On the other hand if workarounds are worked into the system in a flexible way, it could help improve the system (Ibid, 199).

7.2.1.1 Exception Handling

It is hard to tailor a system to meet the demands of irrational human behavior and the different work practices of all the individual workers. People improvise and have their own ways of handling exceptions. Work description and actual work processes are two different things, and a system needs to be flexible and take this into account (Grudin 1994b, 98).

As there are some shortcomings in Magnolia, some exception handling must be done. The example of pre-dating a cancelled subscription in connection with moving shows that some work has to be done in the old system. The users themselves made some individual efforts to handle these exceptions, and some were official messages from the management to do certain processes in a certain way.

The old system is still kept in these kinds of cases. The plan for the future is to complete the system and implement support for reported exceptions. They have also had a focus on

discovering these by getting feedback from the users. Magnolia in itself is so full of functionalities that I deem it flexible to handle these cases, even though it needs some improvement.

Grudin's solutions are based on flexibility to handle exception cases and avoiding the myth of how things are supposed to work. Systems will not work if they are dependent on management urging their reluctant users (Grudin 1994b, 98-99). In the case of Magnolia both users and management seem set on handling exceptions, and the workers do not seem reluctant in using the system even though there were initial problems. One user reported that she was more negative than positive on using Magnolia, but even she saw the benefits of using it and was hoping for improvements.

7.2.1.2 Designing for Infrequent Use

Not all groupware functions will be constantly used. By relying on extra applications to do certain tasks may ensure that it is not used (Grudin 1994b, 99). Magnolia has solved that potential problem by implementing these functions in one piece of software. There are exceptions, because they are also using e-mail, calendars, intranet pages and information screens.

The most work related functions are still implemented in Magnolia. E-mail is used for messages concerning meetings and general information, and also for social non-work interaction. The same goes for the intranet which is divided into work related and social. The information screen is present in the work environment and used as a reference point. Meetings are compulsory.

My observations and interviews showed me that everybody used e-mail and checked it regularly. When asked about the intranet pages, they told me that it had become part of the daily routines, and that they updated themselves when arriving at work and during their breaks. Some were more enthusiastic than others, but it did not seem to be an issue.

Generally, everything that could be viewed as groupware functionality is integrated and supported in a way that when they need to use it, it is available and they know where to find it. Avoiding negligence of infrequent used standalone applications could be seen in relevance to disturbances and avoiding use of alternatives as warned by Ciborra (Ciborra 1996, 197). If a groupware application is seen in this kind of negative light the use of

alternatives could be more tempting, thus further mangling the groupware's chance of success. This is an example of the negative sides of workarounds.

In any work place there is one or more systems used for the daily routines, systems that is needed to get the work done. By including groupware functionality in these programs one skips the step of introducing additional software that could mean extra work, and lowers the threshold of groupware being used (Grudin 1988, 87).

7.2.1.3 Drifting

I find some evidence about Ciborra's point of drifting, both for the users and the management. The system was unfinished and flawed when released, the acceptance was low and there were huge performance problems. Ciborra says that workarounds and different work practices outside the original goals is not necessarily negative (Ciborra 1996, 198).

The evidence is that the whole process has been centered on the users and taking their views serious. The people working with improving Magnolia are hoping to make it a more process oriented CRM, basing it on user reports and hopefully on management needs. The positive progress of Magnolia could be said to be an effect of drifting. Management needs may be a little behind, but there are at least efforts and hopes to get more inclusion.

The people I interviewed and observed seemed to gradually implement their own work processes. Some of them may be a part of the solution to the performance problems. One of the users told me he had started using a technique of reducing the number of "clicks". He told me he experienced better performance by working this way, which is logical because there are fewer server requests this way.

Imposing this view on all the other users is not the answer, but it may be wise to take it into consideration. One of my management sources told me that they were thinking out ways to reduce the number of clicks and create more processes based on best practice and how much it was used.

Ciborra's view of implementing these work practices to make the system become a more integrated and invisible part of the work life (Ibid, 199) is obviously shared by the people maintaining Magnolia.

7.2.1.4 Non-technical Workarounds

During my observation I noticed that a lot of information was passed around on post-it notes or face to face. The reasons for this are not easily identified in CSCW-literature, but could be seen as a non-technical workaround of the system. On one side it could be viewed as negative, because it implies that functions in the system are being neglected. A message concerning customer follow-ups could be sent by e-mail.

On the other side it is an example of the social dynamics and interaction within the group. It may seem easier and more enjoyable or polite to deliver the message personally. It gets the job done and they get to interact with each other. It could also be evidence of a need and benefit of face to face interaction (Putnam 1995, 70)

7.2.2 Workarounds and Social Capital

I will discuss how workarounds in groupware systems could be related to social capital issues.

7.2.2.1 Interaction

There is an importance of social interaction face to face. This is very relevant in accordance to the usage of technology to facilitate social capital. Both Uslaner (Uslaner 2000) and Putnam (Putnam 1995) describes the strengths in social interaction with peers. Or as in Putnam's example, the importance of the bowling team eating pizza, drinking beer and occasionally engaging in civic conversations in developing social capital even on a small scale (Putnam 1995, 70).

Uslaner's straight forward meaning is that the Internet could never replace the existing, or create new, communities. The arguments seem to go in the direction that the level of trust and togetherness created in social settings like these could not be surpassed by technology or other media.

In EC they put a lot of emphasis on the social, face to face interaction between the employees. There are social gatherings at a high frequency, and the break room was actively used with people playing at the fussball table all the time. Some of these face to face interactions, like walking around delivering notes to each others, are in some way workarounds of the system. Everything that was delivered through these notes or talks could have been done within Magnolia or Outlook.

But I see this as a necessary workaround for creating a better work environment. The most important things that needed a track record and logging of what was being done, was done in Magnolia. The non-technical, social workarounds were more trivial information and provided a reason to move around during off-phone breaks. Interaction between group members is mentioned in McGrath's matrix of company productivity (McGrath 1984), and in addition to this I believe it strengthens the community, trust and feeling of belonging.

7.3 Awareness and Management Involvement

Awareness says something about having knowledge about your surroundings and what the people around you are doing. It is both what you visibly see, and tacit knowledge of your coworkers. I have included the management involvement in this as it is a part of forming the complete picture of the system and its surroundings.

7.3.1 Awareness, Management Involvement and CSCW

I will look at how awareness and management involvement could influence a groupware system.

7.3.1.1 Difficulty of Evaluation

Analyzing groupware has been described as very difficult. There are many social factors coming to play, and each work setting is unique. A work environment consists of many individuals and groups working together, and their interrelation and backgrounds could be factors to consider. Even successes are hard to evaluate. (Grudin 1994b, 100). Each groupware has to be seen in its context and success evaluations from one company do not necessarily apply to another. Success could also be seen as something that is measured over time (Ciborra 1996, 194-195).

It's difficult to create a test group that reflects the different social, motivational, economic and political factors that are involved in a group's performance. Field observations of groups are complex because of this. Not only must groups of people be observed, but group variations, environmental factors that affects the acceptance of the system (education, management buy-in and follow up) is also crucial. It is easier to determine failure or success, than to pin point the factors leading to either one. If we add add-ons, features and different

user interfaces depending on job, background and preferences, the evaluation becomes increasingly difficult (Grudin 1988, 87-88).

In the case of Magnolia, the decision to choose that specific system was based on initiatives from a person who had implemented it in other companies. It is based on a rather flexible CRM system open for specification for the individual companies, Siebel. I don't know any success stories about Siebel in general, besides the advertisements. But it's a widely used system so I'm guessing that there are some success stories.

However, as Ciborra states each groupware case is special and success stories from other companies may not apply. The problems at the start of the project imply some errors in evaluating the needs of the company and the users. On the other hand, as the first after-launch evaluation concluded that the process had been a failure they started evaluating what had gone wrong, and what could be done to improve it. This evaluation was done close with the users and it is now in the process of improving the system according to user and company needs.

As the evaluation and getting the system at an acceptable level process is not completed yet, it is hard to determine if they have been successful in their approach. But Grudin's solution to this challenge is to recognize the fact and gain input from users at all levels (Grudin 1994b, 101). This has been done in EC.

7.3.1.2 Breakdown of Intuitive Decision Making

It is not easy to have a good intuition for creating or selecting groupware. In the same way that it is difficult to evaluate groupware, it is also difficult to take into consideration and understand all the complexities involved in groupware (Grudin, Groupware and Social Dynamics: Eight Challenges for Developers 1994b, 100). Each groupware system has individual needs that need to be understood and best practices do not necessarily apply to all situations (Ciborra 1996, 196-197).

In the case of Magnolia it's hard to determine if there was intuitive decision making or not. The choice of basing it on a well known CRM, Siebel, was based on a belief in best practices but also because a person had experience with implementing Siebel in other companies. In a way this shows some intuition behind the decision of system. But on the other hand it was possibly based on a thought that "if it fitted somewhere else it should fit here".

The resulting problems of getting it deployed in time and lacking stability and functions at go-live shows some evidence that there was a lack of intuition for the system in that precise setting. Grudin talks about everybody being to some degree biased to their jobs and what fits them (Grudin, Groupware and Social Dynamics: Eight Challenges for Developers 1994b, 101). This could be shown in the enthusiasm of the person who spearheaded the introduction of Magnolia, as he had worked with the same kind of system before.

Grudin's solution is not very direct, but that user involvement in development is important. Involving regular users does have its obstacles and challenges, but also the gains of realizing the complexities and issues that could have been omitted. Management should realize the risks of intuition and gain more input from users (Grudin, Groupware and Social Dynamics: Eight Challenges for Developers 1994b, 101).

Concerning Magnolia it was hard to determine at what point the user involvement had started. I got the feeling that it was not during the decision making but at a later stage during the testing and development. At that stage they were active in getting user input, and after the deployment the user input had a lot to say for the maintenance and further development.

In that sense they have shown some good intentions of not basing everything on intuition. Ciborra's views on drifting could also be seen as a way of avoiding bad intuition by both having design goals and being flexible in changing these goals down the line.

7.3.1.3 Social, Political and Motivational Factors

I did not find evidence of any imposing on the social dynamics in the company. If there were, it did not come forth in my interviews or observation.

7.3.1.4 Awareness

The open office environment, team structure, message boards, meetings and informal communication leads to some degree of awareness in EC. One of my interviewees who had worked there for a couple of months felt the company was so big that it was difficult to get a grasp of it during the training in the beginning. But after getting socialised and introduced to the work environment her department at least became more lucid. The complete company structure is not something that you learn in a short while.

The customer support department is situated in a big open office environment and supported by information screens. If there is a crisis or an influx on incoming calls, it is easy to notice and spread the news. In a case where many people are working off-phone and there suddenly is a lot of incoming calls, this is shown on the traffic monitoring screens and a team leader or group leader goes around informing people in case they didn't notice.

The awareness in that sense agrees with the findings in the print industry shop floor, that awareness supports the workflow and makes processes run more smoothly (Bowers and Button 1995, 56). Within the teams at least everybody had some tacit knowledge about who knew what, and where to go with requests. More experienced users knew their way around the whole department.

I found some evidence of awareness leading to increased system acceptance. One person I interviewed had been a dedicated Magnolia hater, but had now ended up being one of the most positive persons in the whole department. She explained it with getting more responsibility and needing to be a good example, but also that she had gained more insight in the organization and the system. She is now a super user with greater insight in "what does what" and the reasons behind procedures. More insight also tells her that the Magnolia team are working with their best intentions in mind.

7.3.2 Awareness, Management Involvement and Social Capital

I will use the awareness term from CSCW in regards to social capital in my data.

7.3.2.1 *Belonging*

Another point that I have distilled from Putnam is creating a feeling of belonging. That you are more likely to contribute to a society you feel you are a part of, or in stronger terms; that you matter. It is related with both civic engagement and interaction. There is a difference in social capital between large, or tertiary, organizations and associated memberships. While you pay your membership fee and possibly read a newsletter in the first, you gain no intimate feeling of belonging and develop no social capital from the membership.

More close-knit social groups award one users contribution in a circular sense, where one contribution increases the social capital and spawn other contributions; which in turn could reflect back to the benefit of the original contributor. Participation is of vital importance to social capital (Putnam 1995), and the users should be encouraged to participate either by

the strength of the network itself or by convincing them to see the strength (Girgensohn and Lee 2002, 139).

It could have base in an ancient notion of humans as "pack animals", but the personal benefits of a strong network should not be overlooked. "*Benefit given, benefit received*" as mentioned in an article about the benefits for researchers with strong networks in relevance to the lectures held (Licamele, et al. 2005).

I felt a positive atmosphere in the work environment, a feeling that everybody knew everybody. Or at least that everybody knew somebody, and by extensions had links throughout the department. The company provided many social activities both in the office and outside. I was told that the social gatherings had a high attendance rate. Some of my interviewees said explicitly that it was an environment that created a lot of trust.

Working this close combined with the social activities and good atmosphere can create an environment where people feel they belong and are being a part of something. Belonging can also increase trust and interaction, and is in my view important for the awareness. Belonging to a group increases the possibility of knowing the individuals and goals of that group, and possibly the group's interaction with other groups.

The people who talked about good social interaction in EC hinted about feeling a part of the group. They felt they were taken care of and that their opinions mattered, as in the case of feedbacks of the system. With that I feel that the sense of belonging strengthens the system acceptance and awareness in the company.

7.3.3 Users vs. Management

Grudin amongst other warned at an early stage about having a discord between user groups, namely management and regular users. The early years of CSCW were filled with examples of badly introduced groupware that was sabotaged by the users (Grudin 1988). He follows this line to a lesser degree in his "Eight Challenges" paper.

In EC I found this to be the most positive side of the whole process. Even if there were some occurrences of lack of communication between the management of different departments, the traditional top-down dissension was not present. It seemed like the management took the concerns of the users seriously, at the same time that the users understood that the

management had their best interests in mind. By being included in many stages of the testing and being listened to after deployment, they saw results.

One user said she almost hesitated to report problems to the system operators because she saw how they also struggled with the same stability and performance problems as the users. And even though some were opposed to the new system, they seemed to understand why it was being introduced and felt that they were well informed about it. The issue of the users versus the management was not present.

8 Conclusion

8.1 Summing up

It is hard to have a very precise conclusion when studying something in a large research field as CSCW. The most interesting thing to do with my case is to watch it further, because the management and developers of Magnolia are very conscious of their responsibilities and challenges. They have already good ideas they want to test out, and the next update to the system is due this May, after this thesis is delivered.

The whole process has been very interesting and it has been very inspiring to investigate an actual system, comparing it to CSCW and social capital theories. I will now sum up my conclusions regarding the three central themes and my final conclusion on the system acceptance and user involvement. Then I will look at further work that could be done on the subject.

8.2 System Acceptance

In one way I can say that the acceptance problem in some way is taking care of itself. The system is in a good trend of increasing performance and implementing the required functionality. At the same time there is in these kinds of companies some labor turnover. Some are part time employed, others are taking a year of studying, others again are moving to new jobs. Those who stay seem to get more responsibility or are promoted.

These factors combined make the outlook good for Magnolia, because my data shows me that new employees are more positive and don't see as many problems as the old

employees. And the pre-Magnolia employees that have been promoted/given more responsibility get a more positive view afterwards. Almost all users had a positive view of the future of Magnolia.

The positive and unifying work environment is also a factor that will contribute to system acceptance when the system has reached a more satisfactory state. If there are no more new issues with the system, I believe that the good community spirit will ensure a stable feeling towards the system. To get the system acceptance to a higher level Magnolia needs to be shaped more to fit the work reality and the users' wishes and be more suited for the intensity and speed of a customer support department.

This must not be used as a cushion, but rather use the whole process as knowledge for future attempts. Regarding the pitfalls of CSCW there was some success and some failure. The failures came during the introduction, as errors both foreseen and unforeseen troubled the launch. In accordance with Grudin's warnings this almost caused a catastrophe, but measures taken afterwards saved it and made it a usable system – if not a success yet.

What I got from social capital data was mostly positive, even though there is a possibility that the collective skepticism made things worse during the introduction. The community feeling was at a high level and the users participating in their society increased trust towards their coworkers.

I feel at least that the social framework for making Magnolia a successful system is in place, but to achieve the best system acceptance the system must be made more stable and decrease the time it takes to do normal procedures.

8.3 Workarounds

The workarounds I saw was similar to Ciborra's talk about drifting (Ciborra 1996, 198-200). Some were introduced from management level; others were taken into use by the users to omit shortcomings of the system. Some of these will possibly go on to be more widespread or implemented into the system. The result of this should be interesting to see. Other workarounds were more illogical and based on user preference, but should not be discarded

as user errors as they didn't necessarily decrease productivity. The negative side is that they show an unwillingness to embrace Magnolia in full.

The extended use of the old system is a danger, but it will at some time in the future be removed from the front offices and at that time Magnolia will hopefully have full functionality. Magnolia looks flexible for fitting it into this work environment, and Grudin's challenges of infrequent use and exception handling is somewhat diminished by how the groupware functions are integrated and the system being flexible and full of functions.

The worker interaction and use of oral messages and post-it notes was for me a positive thing, building up the interaction part of social capital and thus strengthening the community feeling and sense of belonging.

8.4 Awareness and Management Involvement

The management needs to get more involved with a more direct goal in the Magnolia project. From my point of view their involvement had been too vague and unfocused and was a factor in the chaotic launch. They're not the only ones to blame, far from it, but a more strict direction could have ensured better quality. It is evidence of Grudin's views on the intuition for groupware and difficulty of evaluation, but I still see positive moves being done here especially in the phase after go-live.

When it comes to awareness, there were positive signs about people being aware of their coworkers and place in the system, but this could also be improved. Evident by those who had a high degree of awareness, this was not only positive for their skills and know-how, but also for their general "feel" for the system. I will conclude that measures should be taken to ensure and empower awareness.

The downside of this is that there is some degree of turnover and that awareness could be said to be something also gained over time.

8.5 Answering the Question: User Involvement and its effect on System Acceptance

My research question was if an increased focus on user involvement in developing groupware has a positive effect on system acceptance or not.

My quick and straight forward answer would be, yes; an increased focus on user involvement *does* have a positive effect on system acceptance. I take evidence from this from my case study which showed that all the things that went wrong during the introduction was fixed with the help of user involvement. There are of course technical solutions that have increased stability and performance, but the system could have failed during the initial stages if the users hadn't turned around and starting accepting it at an increasing rate.

I will state that the system acceptance level at the moment is satisfactory, even though it could get better. I also see the user involvement view as crucial in reaching this acceptance.

Another interesting aspect of my thesis is that Grudin's eight challenges is a good way to analyze a groupware system – or to take into consideration when developing a groupware system. It is not a fool-proof, step by step instructions and some of the views are somewhat bombastically negative, but I feel that the general ideas are relevant even today.

Using social capital to analyze a groupware system was also interesting, because I believe that the ideas of communities are applicable in every field of research that involves people in groups.

I have tried to be somewhat positive in my conclusions as I see the work done after the introduction as such. It is not a perfect process and there were several things to pick on in the phases from planning to the introduction. My concentration have been more on what has made it better after the terrible start.

8.6 Further work

What is interesting with analyzing groupware is watching it over time. A groupware system evolves over time and matures by use. It would also have been interesting to see the next step of groupware introduction in EC. They are implementing a presence awareness tool

later in the year that will keep track of work schedules, vacations and breaks. It is not as huge and fragile as the Magnolia introduction, but it would have been interesting to watch if they have gained wisdom from the Magnolia process to make this introduction smoother. Maybe similar cases from similar companies would have been informative to compare.

The most important part is perhaps to get the views of the CSCW field heard. There is perhaps an increasing focus on user involvement, but sometimes it feels like there is no structured theory behind it when it comes to companies' actually planning and developing groupware.

It would be interesting to have a larger groupware project that is purely based within CSCW thinking and compare it to other comparable systems to see if there are any differences in approach and result. As I have been within this field of research for only a short time this is especially interesting to me

My feeling is at least that there is still more to learn about groupware introduction processes, and that I will be interested in looking at more examples with a user involvement view. I have a positive belief that this will improve a lot in the future.

Bibliography

Ackermann, Mark, John M Carroll, Giorgio DeMichelis, Marlene Huysman, Barry Wellman, and Volker Wulf. "Communities and Technologies: An Approach to Foster Social Capital?" *CSCW'04*, 2004: 406-408.

Adler, P.S, and S-W Kwon. "Social Capital: Prospects for a new concept." *Academy of Management Journal*, 2002: 17-40.

Blomberg, J, J Giacomi, A Mosher, and P Swenton-Wall. "Ethnographic Field Methods and Their Relation to Design." In *Participatory Design: Principles and Practices*, by D Schuler and A Namioka, 123-156. Hillsdale, N.J.: Erlbaum?, 1993.

Bourdieu, Pierre. "Ökonomisches Kapital, kulturelles Kapital, soziales Kapital." *Soziale Ungleichheiten (Soziale Welt, Sonderheft 2)*, 1983: 183-198.

Bowers, J., and G. & Sharrock, W. Button. "Workflow from Within and Without: Technology and Cooperative Work on the Print Industry Shopfloor." *Proceedings of the Fourth European Conference on Computer-Supported Cooperative Work*, 1995: 51-66.

Bowers, John, and David Martin. "Informing Collaborative Information Visualisation Through an Ethnography of Ambulance Control." *Proceedings of the Sixth European Conference on Computer Supported Cooperative Work, 12-16 September 1999*, 1999: 311-330.

Ciborra, Claudio U. "What does Groupware Mean for the Organizations Hosting it?" In *Groupware and Teamwork*, by Claudio U Ciborra, 191-209. John Wiley & Sons Ltd, 1996.

Dourish, P., and V. Bellotti. "Awareness and coordination in shared workspaces." *Proceedings of the 1992 ACM conference on Computer-supported cooperative work*, 1992: 107-114.

Ellis, C A, S J Gibbs, and G L Rein. "Groupware - Some Issues and Experiences." *Communications of the ACM vol. 34 no. 1*, 1991: 38-58.

Flick, Uwe. *An Introduction to Qualitative Research*. London: Sage Publications Ltd; Second Edition, 2002.

Girgensohn, Andreas, and Alison Lee. "Making Web Sites Be Places for Social Interaction." *CSCW'02*, 2002: 136-145.

Grudin, Jonathan. "Computer-Supported Cooperative Work: History and Focus." *IEEE Computer Volume 27, Issue 5*, 1994a: 19-26.

Grudin, Jonathan. "Groupware and Social Dynamics: Eight Challenges for Developers." *ACM Vol 37 No 1*, 1994b: 93-105.

Grudin, Jonathan. "Is HCI Homeless? In Search of Inter-Disciplinary Status." *Interactions: Volume 13 , Issue 1 (January + February 2006)*, 2006: 54-59.

Grudin, Jonathan. "Why CSCW Applications Fail: Problems in the Design and Evaluation of Organizational Interfaces." *ACM (ACM)*, 1988: 85-93.

Huijboorn, Noor. "Social Capital and ICT Adoption in the Public Sector." *Proceedings of the 8th Annual international Conference on Digital Government Research: Bridging Disciplines & Domains (Philadelphia, Pennsylvania, May 20 - 23, 2007)*. *ACM International Conference Proceeding Series*, vol. 228, 2004: 140-147.

Huysman, M, and V Wulf. *Social Capital and Information Technology*. Cambridge, Massachusetts: The MIT Press, 2004.

Huysman, M, and V Wulf. "The Role of Information Technology in Building and Sustaining the Relational Base of Communities." *The Information Society*, 2005: 81-89.

Katz, and Rice. *Social Consequences of Internet Use*. Boston: The MIT Press, 2002.

Licamele, Louis, Mustafa Bilgic, Lise Getoor, and Nick Roussopoulos. "Capital and Benefit in Social Networks." *LinkKDD'05*, 2005: 44-51.

McGrath, J. E. *Groups: Interaction and Performance*. Prentice Hall, 1984.

McLean, Scott L., David A. Schultz, and Manfred B. (eds) Steger. *Social Capital: Critical Perspectives on Community and Bowling Alone*. New York and London: New York University Press, 2002.

McNally, Mark, Majella Bannon, John Greaney, Aisling Hickey, Marian McDonnell, and Mark Riordan. "The Effects of a Wireless On-Line Community Network on Social Capital." *ACM*, 2005: 215-218.

"Moscito Project Proposal."

Nardi, Bonnie A. "The Use of Ethnographic Methods in Design and Evaluation." In *Handbook of Human-Computer Interaction*, by M Helander and T.K Landauer, 361-366. Elsevier Science B.V, 1997.

Putnam, Robert D. *Bowling Alone - The Collapse and Revival of American Community*. New York: Simon & Schuster Paperbacks, 2000.

Putnam, Robert D. "Bowling Alone: America's Declining Social Capital." *Journal of Democracy* 6:1, 1995: 65-78.

Uslaner, Eric M. "Social Capital and the Net." *ACM vol 43 no 12*, 2000: 60-64.

Wilson, P. *Computer Supported Cooperative Work: An Introduction*. Kluwer Academic Pub, 1991.